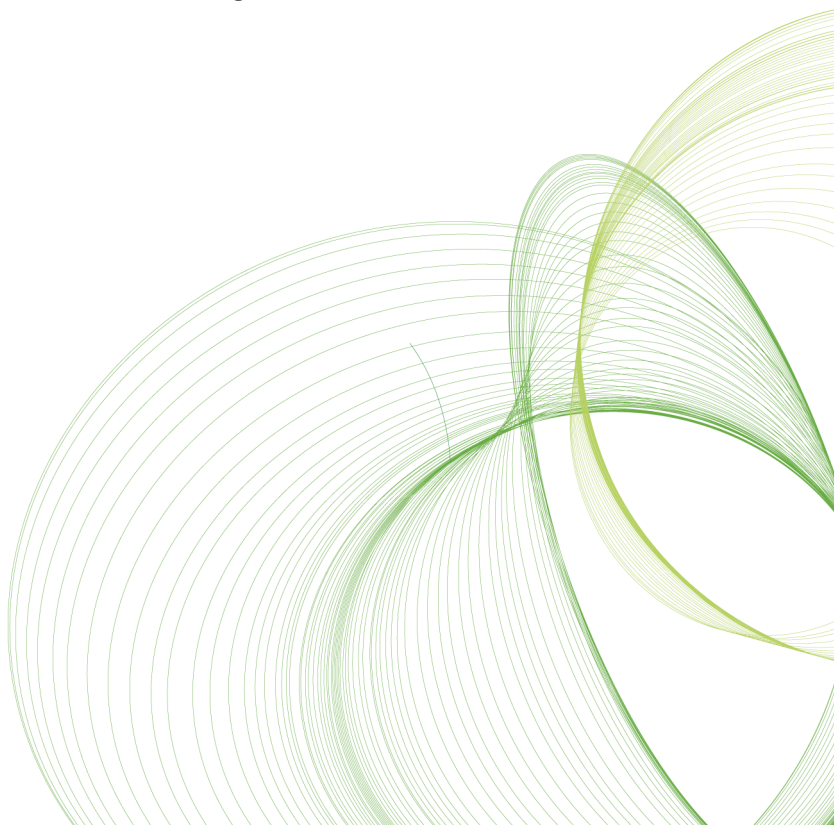




Designer II

September 2009 Release

QlikView Version 9.00 English



Copyright © 2009 QlikTech International AB, Sweden.

Under international copyright laws, neither the documentation nor the software may be copied, photocopied, reproduced, translated or reduced to any electronic medium or machine-readable form, in whole or in part, without the prior written permission of QlikTech International AB, except in the manner described in the software agreement.

Qlik®Tech and Qlik®View are registered trademarks of QlikTech International AB.

Microsoft, MS-DOS, Windows, Windows NT, Windows 2000, Windows Server 2003, Windows Server 2008, Windows XP, Windows Vista, SQL Server, Excel, Access, Visual Basic, Internet Explorer, Internet Information Server, Visual C++, Visual Studio and MS Query are trademarks of Microsoft Corporation.

IBM, AS/400 and PowerPC are trademarks of International Business Machines Corporation.

Firefox is a trademark of the Mozilla Foundation.

Apple, iPhone, iPod Touch, Safari and MacOS is a trademark of Apple Corporation.

BlackBerry is a trademark of Research In Motion.

Original release: May 2009

This release: September 2009

CONTENT

1 INTRODUCTION 5

Installing the Course Materials	5
Program versions	5
Text formats	5

2 MULTIDIMENSIONAL CHARTS, PIVOT TABLES 9

Straight Table with Multiple Expressions	9
Pivot Tables with Multiple Dimensions	10
Grouped bar charts	13
More about Pivot Tables and Straight Tables	15
Using a Dropdown Selection List	15
Using Visual Cues	16
Summary – Comparing Tables	16

3 ADVANCED CHARTS, TABLES & OBJECTS 23

Search Objects	23
Clickable URLs	24
Tree View List Box	26
Mini Charts	27
Trellis Chart	28

4 ADVANCED CALCULATIONS IN SHEET OBJECTS 39

Set Analysis	39
Dollar-Sign Expansion	43
AGGR Function	44

5 GAUGES 51

Why Use a Gauge Chart?	51
Types of Gauges	53
Building Gauges	54

6 BUTTONS AND ACTIONS 61

Buttons in QlikView	61
Creating and Formatting a Button	61
Actions	63
The Line/Arrow Object	65

7 OTHER SHEET OBJECTS 71

The Slider/Calendar Object	71
The Input Box	72
The Bookmark Object	72
The Custom Object	73
The Other Chart Types	73

8 ADDITIONAL EDITING IN THE LAYOUT 79

Dynamic Chart Backgrounds	79
Title Settings	81
Fast Change Chart Type	81
Legend Settings	81
Reference Lines / Text in Chart	83
Caption Settings	83
Using Layout Themes	84

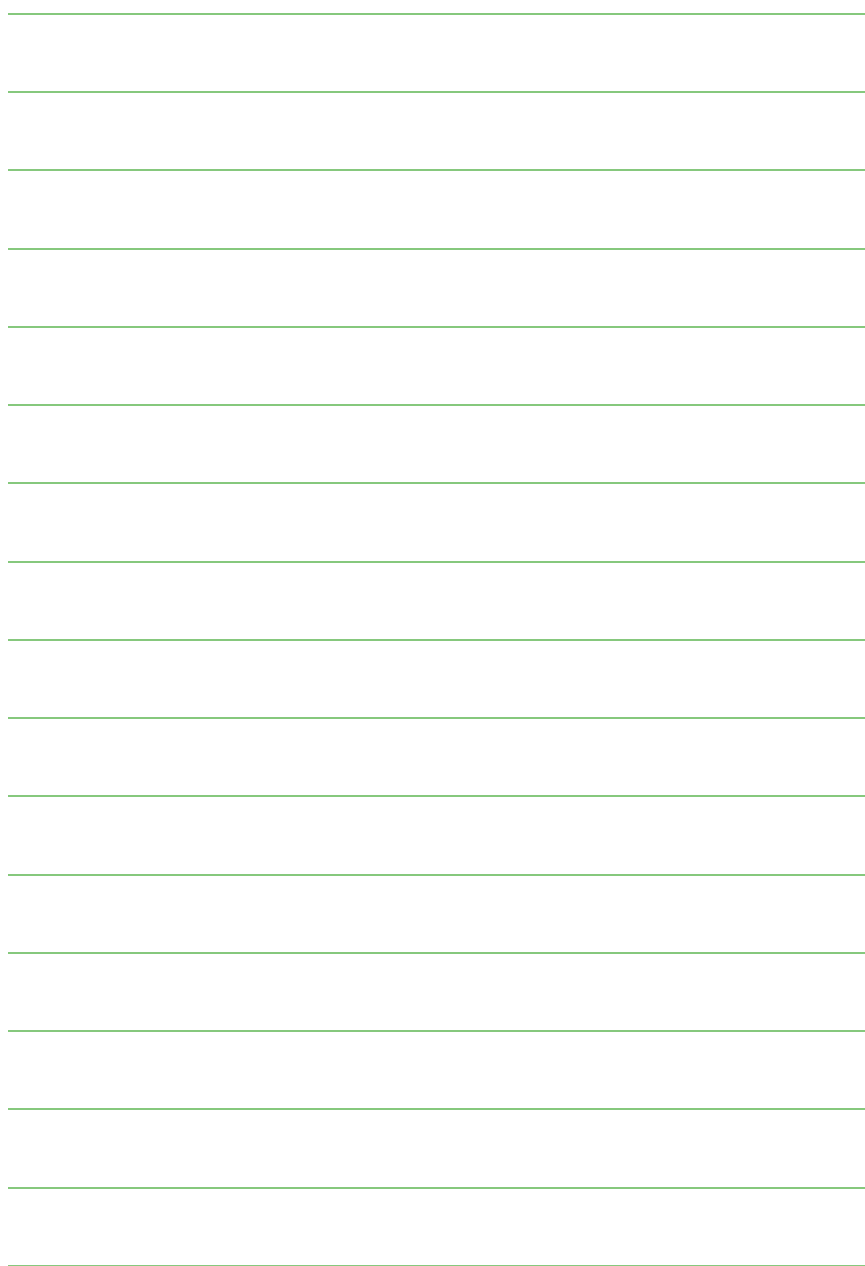
9 OTHER USES OF EXPRESSIONS 89

Calculated Labels and Dynamic Text in Text Objects	89
Conditional Functions	90
Chart Attribute Expressions	91
Color Expressions	92
Calculated Fields in List Boxes	93
Calculated Dimensions in Charts	94
The Expression Overview Dialog	94

10 REPORTS AND BOOKMARKS 101

Introduction – Reports	101
QlikView Report Types	102
Managing Reports	103

Creating and editing reports	104
Introduction – Bookmarks	106
Bookmark Types	107
Managing Bookmarks	108
Creating Bookmarks	110
Mailing Bookmarks via QlikView Server	110
E-Mail as Attachment, Other Options	111
11 BUSINESS CASE WORKSHOP	115
Exercise Scenario	115
Data Sources	119
Application Requirements	120
Final Comments on Building With QlikView	121
12 REVIEW AND CONCLUSION	123
Review	123
Collaboration	123
Conclusion	123



1 INTRODUCTION

Many students will come to this course having already taken QlikView classroom training. This chapter covers formatting and conventions used in the course manual.

Installing the Course Materials

The course materials will self-extract from the file into the default directory

C:\QlikViewTraining\DesignerII

Make a Windows shortcut to this folder and place it on your desktop.

Also make a Windows shortcut to the documentation folder and place it on your desktop.

C:\Program Files\QlikView\Documentation

Program versions

This course was created using the English version of QlikView 9.00 running on WindowsXP. If other operating systems or languages are used, minor differences may be noted in the visual appearance of windows and dialog boxes.

Text formats

Exercises and actions to be completed by you, the student, will be set-off with a logo, as you see, below:



Exercise/Do:

This is a sample of instructions you would see to complete an exercise containing a sequence of steps.

- 1 Click on the **Start** button
- 2 Locate the QlikView icon
- 3 Click on the QlikView icon to launch the program

All commands, as well as all names of menus, dialogs and buttons are in the following font style: **File - Open**

All names of list boxes, graphs and specific data in list boxes, etc. are in the following font style: *Country*

All file names are in the following font style: **QlikViewCourse.qvw**

Tips and Notes are outlined in a highlighted box, as you see below:

This sample sentence is used to illustrate important points in the text, tips and notes to consider as you complete the course materials



EXERCISE – SETTING UP YOUR ENVIRONMENT

Do:

- 1 Navigate to the c:\QlikViewTraining\DesignerII\Chapter2 directory and open the **QVDesignerII_Chapter1.qvw** file.
- 2 Save a copy of the file to preserve the original in case you want to start again from the beginning later. Do this by using the **File | Save As** command.
- 3 In many environments, consideration should be given to the amount of screen real-estate a QlikView application can use. Some organizations might have standards around the display resolution to be used when deploying applications. QlikView provides tools to address these requirements. Explore the **View | Resize Window** command to set the default value

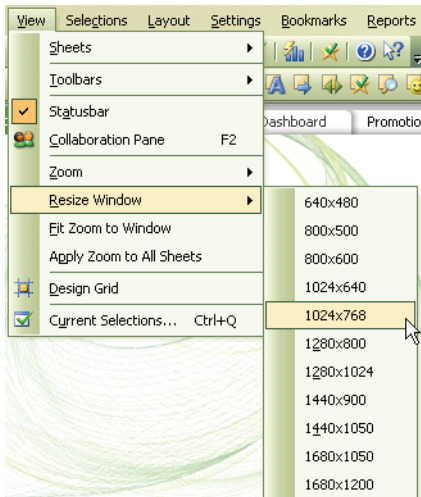


Figure 1. Use **Resize Window** to set the desired resolution

- 4 You can also manipulate the size of individual sheets using the **View | Zoom** command and **Apply Zoom to All Sheets**. Experiment with the **Zoom** command in the **QVDesignerII_Chapter1.qvw** file. Remember, you can always return to the original settings by selecting **Zoom | 100%** for any sheets you have modified, or you can select **Apply Zoom to All Sheets** to set every sheet to the level of the current sheet.



- 5 Take this opportunity to activate the toolbars of your choice for the class. At minimum, you should use the **Standard** and **Navigation** toolbars. Note, too, that the toolbars can be moved, floated and docked by clicking and dragging on the dotted line to their far left. Hold down the left mouse button and drag to any position you like. The toolbars can be docked to any side of the QlikView application window. All toolbars are completely customizable and may contain any of the available command buttons. Experiment with toolbars to obtain a good working environment. Obviously, you can change it later, if you so desire.



Figure 2. Moving a toolbar by clicking and dragging on the dotted line to the far left

- 6 Go to the **Settings | User Preferences** menu and navigate to the **Save** tab. Configure this tab for maximum backup by checking the settings, as below:



Note: Feel free to experiment with these settings. Remember that your original exercise qvw file (QVDesignerII_Chapter1.qvw) can be used if you have problems and want to start fresh. Just be sure if you open it to save it immediately with another name, as instructed in the exercises.

2 MULTIDIMENSIONAL CHARTS, PIVOT TABLES

Objectives

- Learn about multidimensional charts in QlikView
- Show examples of straight tables, pivot tables and grouped bar charts
- Use the right chart object to make your QlikView application easier to use

Multidimensional charts contain, as the name implies, more than one dimension. In the previous chapter, we worked with either *Country*, *Category* or *Shippers* in the charts we created, but never more than one of them at a time.

In a multidimensional chart, we might combine *Year* (the time dimension) with *Country* to analyze sales from year-to-year. This type of chart creation will be presented now.

We will also look at an as yet unexplored chart-type, the pivot table.

Straight Table with Multiple Expressions

We will begin with a straight table showing *Total Sales* based on the *Country* dimension found on the *Basic Objects* sheet.

Straight Table	
Country	Sales
	13,321,238
Germany	3,947,735
Mexico	695,819
UK	1,216,681
Sweden	404,404
France	1,285,654
Spain	203,964
Canada	389,493
Argentina	84,037
Switzerland	101,367
Brazil	930,521
Austria	408,404
Italy	95,202
Portugal	97,982
USA	1,421,228
Venezuela	498,457
Ireland	924,285
Belgium	110,569
Norway	1,706
Denmark	418,967
Finland	50,136
Poland	34,626

Figure 1. The Total Sales by Country straight table

In the following steps, we will add an expression to show the average deal size:

- 1 Open the **Chart Properties: Expressions** page.
- 2 Click the **Add** button to create a new expression. The **Edit Expression** dialog appears on top of the **Chart Properties** dialog.
- 3 Create the expression $\text{Sum}(\text{LineSalesAmount}) / \text{Count}(\text{OrderIDCounter})$, click **OK** to get back to **Chart Properties: Expressions** and label the expression *Avg Deal Size*.
- 4 Continue to the dialog page **Number** and edit the number formatting for the new expression to **Fixed to 0 Decimal**.
- 5 Click the **OK** button to complete the table:

Straight Table with 2 Expressions		
Country	Sales	Avg Deal Size
	13,321,238	2,027
Germany	3,947,735	2,616
Mexico	695,819	2,148
UK	1,216,681	2,514
Sweden	404,404	1,813
France	1,285,654	2,267
Spain	203,964	1,091
Canada	389,493	1,630
Argentina	84,037	808
Switzerland	101,367	2,253
Brazil	930,521	1,835
Austria	408,404	1,908
Italy	95,202	639
Portugal	97,982	1,324
USA	1,421,228	1,804
Venezuela	498,457	1,351
Ireland	974,785	2,458

Figure 2. A straight table with two expressions

Pivot Tables with Multiple Dimensions

Suppose that you need a table showing sales figures divided per country, but also subdivided per year and quarter. The result can be shown in a multi-dimensional pivot table:

- 1 Select *USA* and *Germany* in the list box *Country* and lock your selection.
- 2 Clone the straight table from the previous exercise, minimize one and continue working with the other.
- 3 Transform the table into a pivot table (**Chart Properties: General**).
- 4 Continue to the dialog page **Dimensions** and add *Year* and *Quarter* to the **Used Dimensions** list.

- 5 We will only use the expression *Sum(LineSalesAmount)*. Continue to the dialog page **Expressions** and deactivate the second expression. The expression is still available if you need to enable it later on, but for the moment it will not appear in the pivot table:

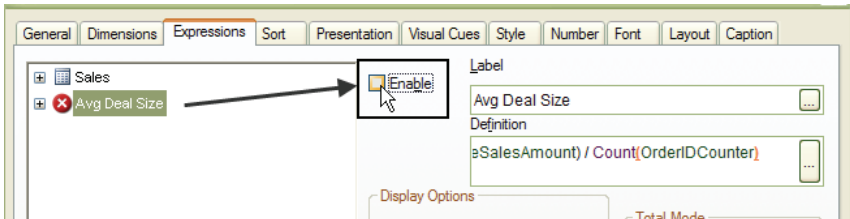


Figure 3. Disabling the expression Avg Deal Size

- 6 Continue to the dialog page **Presentation** and make sure that the option **Show Partial Sums** (subtotals) is checked for the three dimensions *Country*, *Year* and *Quarter*.
- 7 Click **OK** to complete the chart. After some adjustments to the column width, your pivot table should look something like the next picture:

Pivot Tables w/Multiple Dimensions			
Country	Year	Quarter	Sales
Germany	2004	Q2	71,022
		Q3	66,990
		Q4	49,588
		Total	229,058
	2005	Q1	139,739
		Q2	175,512
		Q3	241,974
		Q4	305,080
		Total	862,305
	2006	Q1	356,052
		Q2	288,890
		Q3	459,735
		Q4	382,641
		Total	1,487,317
	2007	Q1	367,029
		Q2	274,334
		Q3	282,598
		Q4	248,422
		Total	1,172,383
	2008	Q1	175,307
		Total	175,307
	Total		3,947,735
USA			1,421,228
Total			5,368,963

Figure 4. The finished pivot table



It is easier to compare the sales in USA and Germany for each period, if we move the dimension *Country* to the very left. The most convenient way to do this is by the drag-and-drop method. To undo this change you can use **Undo Layout** in the toolbar or simply drag the column back to its original place.

You can also try to alter the way the dimension *Quarter* is drawn in the table, so that you get a cross table. You do this by clicking in the column header and dragging it above the *Sales* column.

Pivot Tables w/Multiple Dimensions					
Country	Year	Q1	Q2	Q3	Q4
Germany	2003	-	3,552	11,697	6,11
	2004	41,458	71,022	66,990	49,58
	2005	139,739	175,512	241,974	305,08
	2006	356,052	288,890	459,735	382,64
	2007	367,029	274,334	282,598	248,42
	2008	175,307	-	-	-
	Total	1,079,585	813,310	1,062,993	991,84
USA		307,164	308,195	430,775	375,09
Total		1,386,749	1,121,505	1,493,769	1,366,94

Figure 5. The pivot table as a cross table

The way we allow fields to change positions or rotate the plotted dimensions is called pivoting, hence the term “pivot table”.

There is yet another practical feature of the pivot table, namely the possibility to expand or hide parts of the table. To this end, we use the small  or  buttons. Assuming that we are only interested in detailed statistics for the last two quarters of 2006, the following table gives the answers:

Pivot Tables w/Multiple Dimensions			
Year	Quarter	Country	Sales
2003			34,550
2004			459,827
2005			1,240,186
2006	Q1		451,481
	Q2		402,678
	Q3	Germany	459,735
		USA	151,385
		Total	611,120
	Q4	Germany	382,641
		USA	92,123
		Total	474,764
Total		1,940,043	
2007			1,478,623
2008	Q1		215,734
	Total		215,734
Total			5,368,963

Figure 6. A pivot table where some dimension values have been collapsed

8 Unlock and clear your selection.

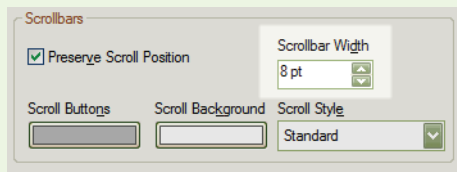
As you can see, the pivot table may already be used for reporting purposes simply by taking advantage of some of the export and print capabilities available directly from the **Object** menu. Using the QlikView Report Editor is covered later in this course.

Grouped bar charts

We will now return to the straight table we minimized earlier and convert it to a bar chart to display the four countries with the best sales figures.

- 1 Begin by restoring the minimized table and convert it to a bar chart. This is done in the **Chart Properties: General** dialog.
- 2 On the **Expressions** page under **Display Options** check **Numbers on Data Point** for the second expression *Avg Deal Size*.
- 3 Also, make sure that the **Chart Subtype - Grouped** is selected under **Chart Properties: Style**.
- 4 Go to the dialog page **Presentation**, change the **Max Visible Number** of bars to 4 and check the option **Show X-Axis Scrollbar**.

Tip: You can adjust the width of the X-Axis Scrollbar by clicking on the Scrollbar width selector in the Layout page of the object, as below —



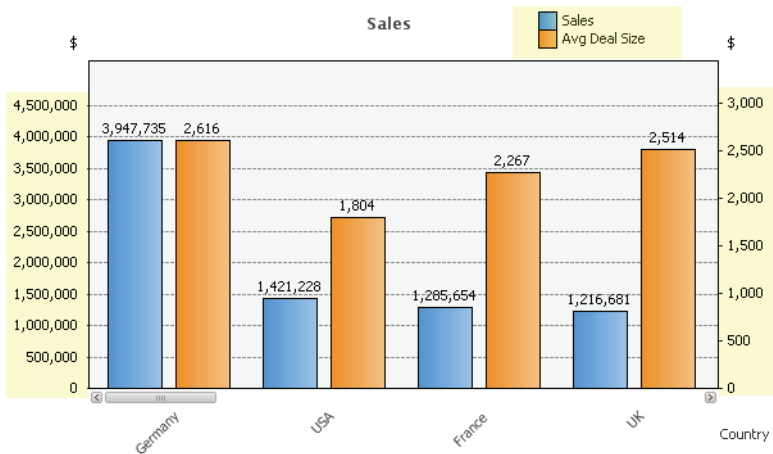


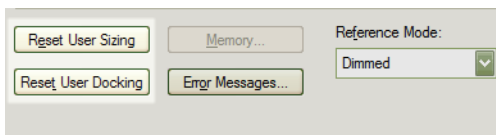
Figure 7. A grouped bar chart with two expressions

Both expressions are now plotted against the same scale on the y-axis. By determining a second scale to be plotted to the right of the chart, we make the chart far more useful for comparing the figures:

- 5 In the dialog page **Axes** you mark the expression *Avg Deal Size* and check the option **Right (Top)** under **Position**.

The picture in Figure 7 shows the chart after a few additional layout changes. Some of these are examples of relatively advanced layout modifications. Here is a very brief guideline:

- Apply the style for simple bars (**Chart Properties: Style**)
- Change the distance between bars and groups (**Chart Properties: Presentation**)
- Change the docking of chart components. Clicking a chart and then pressing CTRL + SHIFT triggers the chart edit mode where individual components of the chart can be moved and resized. Such changes can be restored by pressing the appropriate Reset button under **Chart Properties: General**.



- Display the numbers no longer by 1000 \$ but only by \$ (remove the **Thousand Symbol** in the **Chart Properties: Number** dialog)

More about Pivot Tables and Straight Tables

Let us take a moment to compare the pivot and straight table.

CategoryName	ProductName	Sales	Quantity	CategoryName	ProductName	Sales	Quantity	
Men's Clothes	Atlas Lussekofta	\$89,256.82	2956	Men's Clothes	Atlas Lussekofta	\$5,368,963.09	170448	
	Bow tie	\$35,184.88	4549		Atlas Lussekofta	\$89,256.82	2956	
	Desperado Jeans	\$25,839.19	932		Bow tie	\$35,184.88	4549	
	Lenin Jeansshorts	\$54,770.30	3014		Desperado Jeans	\$25,839.19	932	
	Mi2 Trousers	\$65,994.74	3595		Lenin Jeansshorts	\$54,770.30	3014	
	O-Man Underwear	\$4,472.74	804		Mi2 Trousers	\$65,994.74	3595	
	Rossi Bermuda Shorts	\$33,002.90	4103		O-Man Underwear	\$4,472.74	804	
	Samba Soccer Socks	\$8,848.32	2149		Rossi Bermuda Shorts	\$33,002.90	4103	
	US-Master Jeans	\$61,008.83	2162		Samba Soccer Socks	\$8,848.32	2149	
	Total	\$378,378.72	24264		US-Master Jeans	\$61,008.83	2162	
Women's Clothes	Charrell Shirt	\$30,754.10	1519	Women's Clothes	Charrell Shirt	\$30,754.10	1519	
	Halter Dress	\$1,252,268.31	3241		Halter Dress	\$1,252,268.31	3241	
	Jack Flash Dress	\$194,991.64	3184		Jack Flash Dress	\$194,991.64	3184	
	Langoste Shirt	\$1,507.55	82		Langoste Shirt	\$1,507.55	82	
	Le Baby Dress	\$169,668.02	2058		Le Baby Dress	\$169,668.02	2058	
	Minnki Palsi	\$18,171.86	296		Minnki Palsi	\$18,171.86	296	
	Oklaba Skin Jackets	\$168,890.49	2251		Oklaba Skin Jackets	\$168,890.49	2251	
	Oyaki Kimono	\$36,062.65	3122		Oyaki Kimono	\$36,062.65	3122	

Figure 8. Comparing a pivot table to a straight table

Note the following characteristic differences between the two table types:

- In a pivot table, the data are grouped by field values. In a straight table there is no grouping.
- In a pivot table, partial sums and total sums can be displayed for each group. The straight table can only show the total sum (usually displayed at the top of the table).

Considering the fact that the pivot table is such a versatile tool and that the straight table has the above limitations, why should we bother using the straight table at all? Well, the straight table offers excellent possibilities for sorting, just like the table box.

You may sort the straight table according to the specified sort order by any column (dimension or expression column), just by double-clicking the column header.

Alternatively, you right-click the column and choose **Sort** from the object menu. These sorting capabilities help you to answer questions such as, "What products sell best?"


In addition, it is sometimes easier to export data to Excel or other formats using a simple straight table.

Using a Dropdown Selection List

It is generally highly inconvenient to search for specific field values in a table, simply because tables will quickly grow very tall when every combination of the values is plotted. To complicate matters, we cannot make text

searches directly in a column. To get around these two obstacles, we can add something called **Dropdown Select** capability to any dimension column of pivot tables, straight tables and table boxes:

- 1 Right-click the pivot table, then go to the dialog page **Properties: Presentation**.
- 2 Select *Country* under **Dimensions** and check **Dropdown Select**. Finish with **OK**.
- 3 Repeat the above procedure for the straight table.

A new icon  is added to the left of the column caption. You can click the icon to select one or several products directly in the list or you can even make text searches, exactly as you would do it in a list box.

Using Visual Cues

Pivot tables and straight tables can display the expression values of any column with color-coding. Using the straight table from the previous example, we will now take a closer look at this feature:

Let us assume that we wish to review our sales by country and would like to identify average deal sizes above and below a certain threshold.

- 1 Open the **Chart Properties** dialog of the previously created straight table and go to the **Visual Cues** page.
- 2 Check the expression *Avg Deal Size* and enter the values **1800** in the edit box **Upper** and **1000** in the edit box **Lower**. Check the **Bold** option for these two categories. Finish with **OK**.
- 3 Notice how much easier it has become to determine the relative success of the different county sales teams.

Summary – Comparing Tables

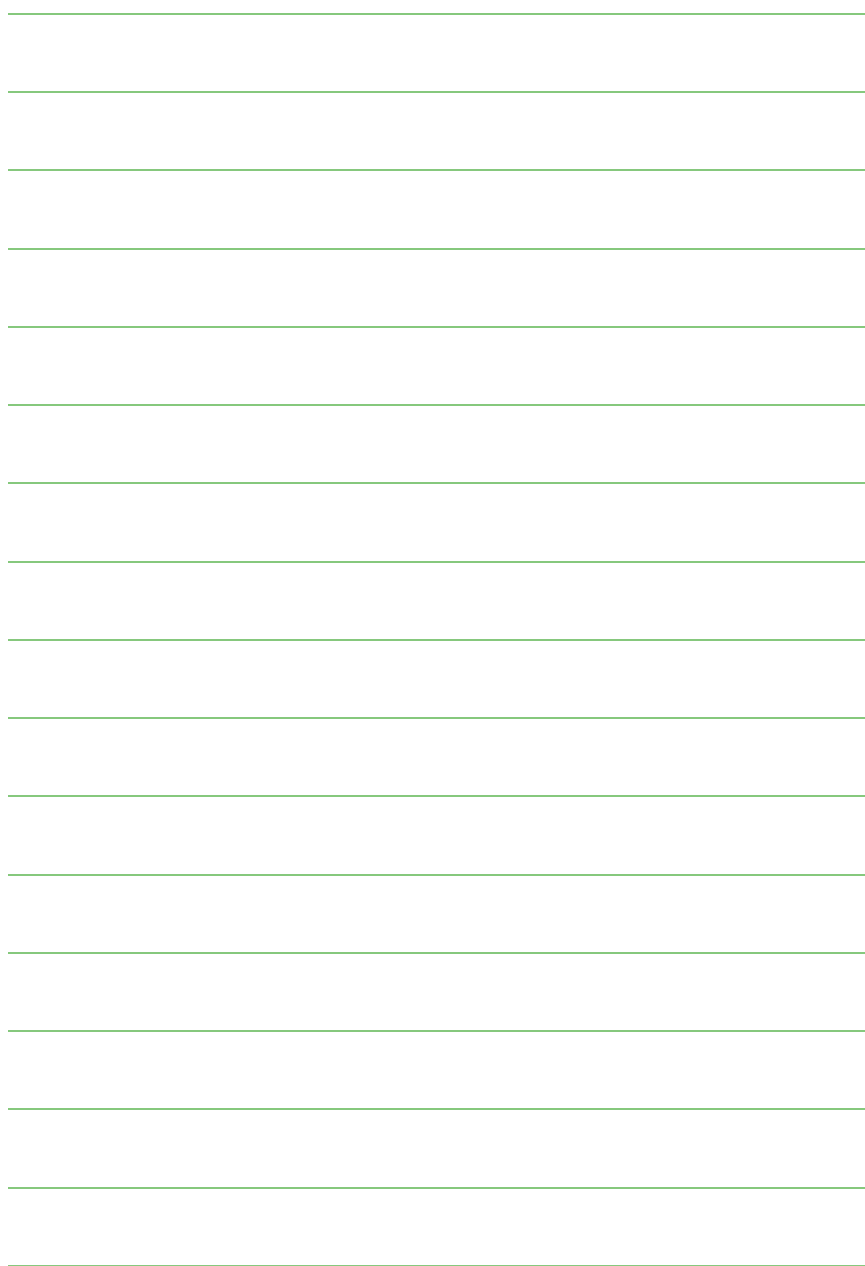
Table boxes, pivot tables and straight tables may look similar, but they are distinctly different object types. Each deals with the problem of presenting data in table form in its own way, as summarized below.

	Pivot Table	Straight Table	Table Box
Type of Sheet Object	Chart	Chart	Table box
Type of data in the table	Dimension + expression values *	Dimension + expression values	Field values (dimensions)
Making selections in field?	Dimension values only	Dimension values only	Yes

Sorting of values?	Limited to changing the settings for dimension values in Properties: Sort.	Yes, dimension + expression values	Yes
Quick sorting in columns?	No	Yes, dimension + expression values	Yes
Calculations/expressions?	Yes ^a	Yes	No
Grouping of data?	Yes	No	No
Partial sums?	Yes	No	No
Total sums?	Yes	Yes	No

a. Pivot Tables may also be used without any expressions to show a hierarchy of fields.

Now that you have been working with charts for a while, it is time to test how much you have learned. In the following exercises, you will create four different charts. Feel free to continue to build objects on your *Workspace* tab.





EXERCISE – MULTIDIMENSIONAL CHARTS

Do:

- 1 Navigate to the c:\QlikViewTraining\DesignerII\Chapter2 directory and open the **QVDesignerII_Chapter2.qvw** file.
- 2 Save a copy of the file to preserve the original in case you want to start again from the beginning later. Do this by using the **File | Save As** command. There is also a QlikView file ending in “_Solution.qvw” containing the completed exercise for your reference.
- 3 Make a line chart showing the sales (Hint: use the “sum(LineSalesAmount)” to calculate sales) per salesperson over time. Use the drilldown group *Drilldown Time* as first dimension and *SalesPerson* as second dimension. *Drilldown Time* is a hierarchical field group that is already defined in the document. You may activate the **Smooth Line** option in the **Chart Properties: Expressions** page to make the chart look nicer. You can also try out some of the formatting options for line charts in the **Chart Properties: Presentation** dialog. When the chart is finished, select a single year to see how the drilldown works. More information about Groups, how to create and manage them, will be covered later in this course.
- 4 Create a pivot table showing the sales per salesperson, product group (*CategoryName*) and year. Let it display partial sums for all dimensions and apply the style **Pivot 1**. Rotate the table so that salespersons and product groups are displayed in vertical columns and years are displayed along the horizontal axis.
- 5 Create a straight table showing the following three expression values over the years: sales total, number of orders and the number of customers (field *CompanyName*). As you can see in the finished table, two companies made it into our database without having any orders or sales assigned to them. Maybe they placed an order and cancelled it afterwards. If you want to remove these values from the table you can check the option **Suppress when Value is Null** in the **Chart Properties: Dimensions** dialog.
- 6 Create a bar chart showing the sales per salesperson and product group (field *CategoryName*). The style is set in the **Chart Properties: Style page**. Sort both dimensions by the sort expression *Sum(LineSalesAmount)* in the sort order you deem most suitable. This will greatly facilitate the interpretation of the chart (see example, below).

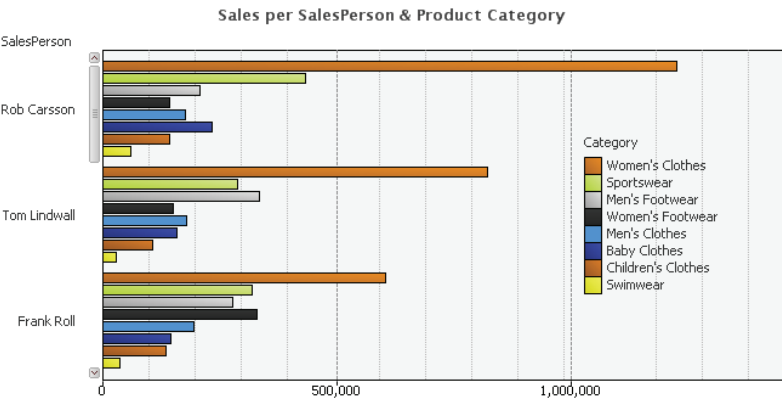


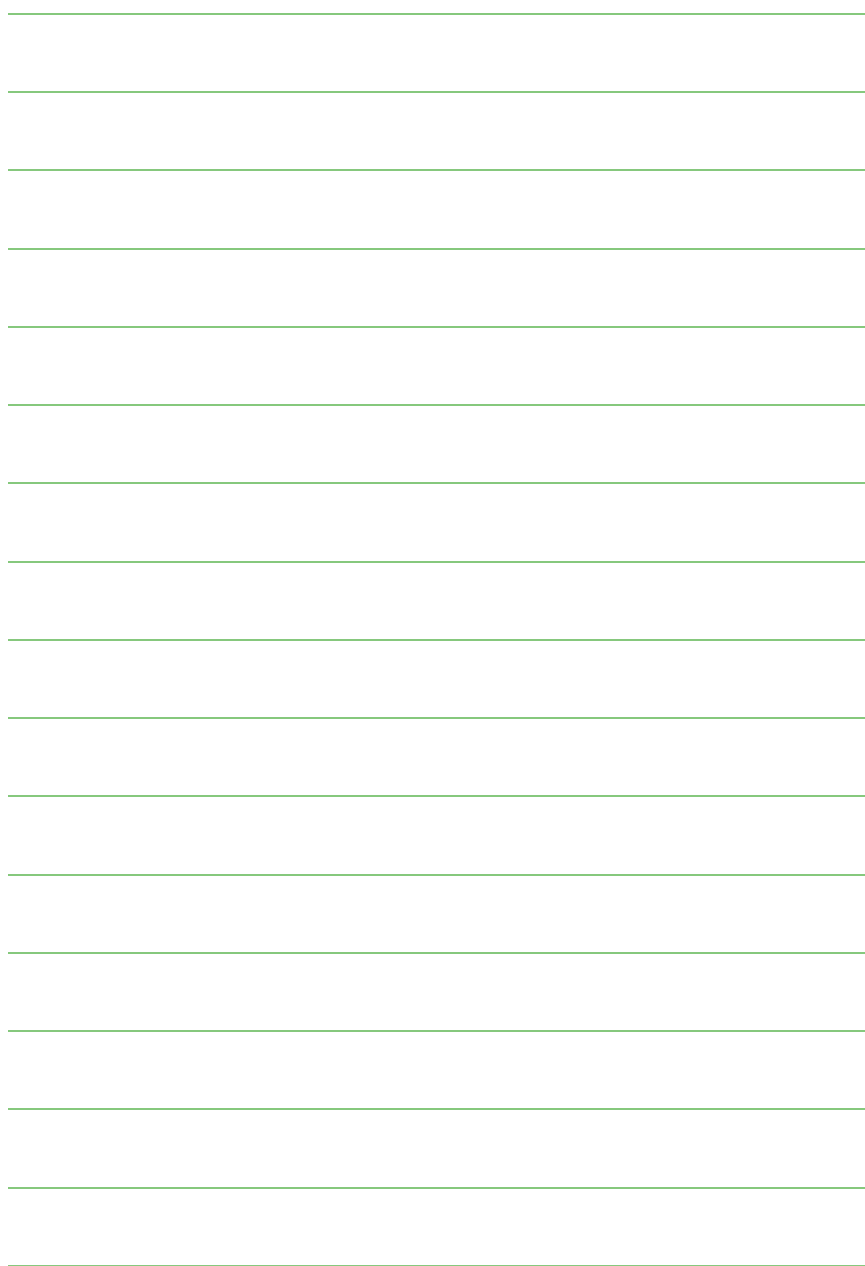
Figure 1. How the final chart in the exercise might look



EXERCISE – OPTIONAL EXTRA, MULTIDIMENSIONAL CHARTS

Do:

- 1 Create a straight table showing sales, cost and margin per product. Use the following expressions: *Sum(LineSalesAmount)* for the sales, *Sum(Quantity)*UnitCost* for the cost and *Sum(LineSalesAmount)-Sum(Quantity)*UnitCost* for the margin. For the expressions *Cost* and *Margin*, check the **Sum of Rows** option under **Total Mode** in the **Chart Properties: Expressions**. Apply the number format **Fixed to 2 Decimals** to all expressions.
- 2 In the previously created straight table add visual cues for margins below \$2,000 and above \$5,000.



3 ADVANCED CHARTS, TABLES & OBJECTS

QlikView offers many advanced capabilities for designing the best display of information. These capabilities tie directly to some of the design best practices explored in the Designer I course.

Objectives

- Learn about advanced charts, tables and objects:
 - Search objects
 - Clickable links
 - Tree view list boxes
 - Mini charts
 - Trellis charts
- Work with these objects in the chapter exercises

Search Objects

Search objects can be used for searching for information anywhere in the document. They are created by choosing new sheet object from the layout menu or from the sheet object menu.

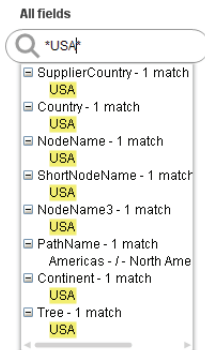


Figure 1. The Search Object in action

How it Works

The search object properties dialog is opened by choosing **Properties** from the **Object** menu. If the properties command is dimmed you probably don't have the privileges needed to perform property changes.

To configure the search options, make selections on the **General** tab. Here you can decide to search all fields, or a specific list of fields.

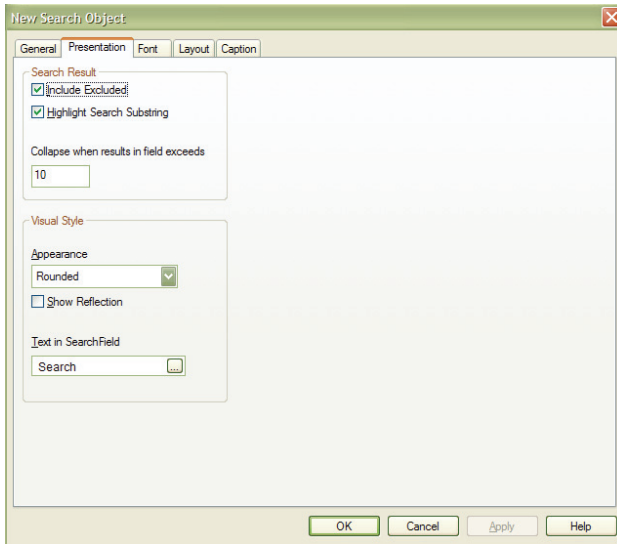


Figure 2. The Presentation tab of the Search Object

The **Presentation** tab gives you the chance to define the look and feel of your search object. You can affect the search result display and the visual style. In the font tab you can specify the font style and size of the text in the search object. The layout tab gives you the chance to specify how the search object should appear on the layout. As with other QlikView objects, this includes settings for shape border and the layer on which the object should reside. Finally, on the **Caption** tab, advanced settings can be made for the caption of the search object.

Clickable URLs

Sometimes it is useful to include a link in a QlikView table, giving users the chance to follow a line of information to a location stipulated by the designer.

How it Works

A representation mode for table cells called **Link** allows you to transform expression cells in straight and pivot tables into clickable links. The normal selection on click behavior is disabled in such cells. You can specify a display text and a URL for the link in the chart expression, and the display text will appear underlined to indicate link status.

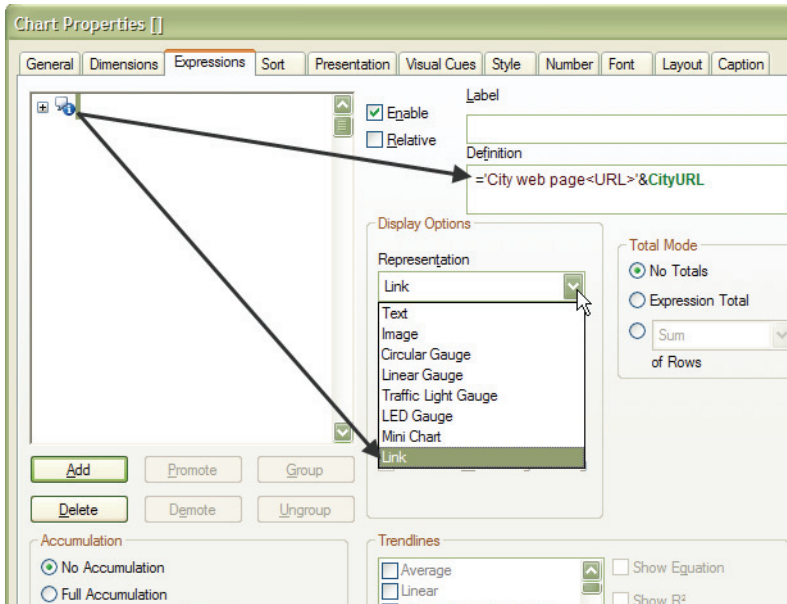


Figure 3. Setting up the expression for a clickable URL in a Straight Table

Note: You will need to load the URL itself into a field value that can be referenced in the **Link** definition, as in

```
LOAD City,
      URL AS CityURL
```

Where the URL value is renamed CityURL and then displayed in the Straight Table.

The resulting straight table:

City	
Aachen	City web page
Albuquerque	City web page
Anchorage	City web page
Barcelona	City web page
Barquisimeto	City web page
Bergamo	City web page
Berlin	City web page
Bern	City web page
Boise	City web page
Brandenburg	City web page

Tree View List Box

There are times when you might like to see a hierarchy displayed within a list box selection, perhaps to better understand the relationships in the data, or simply to fine tune a selection more quickly.

When displaying a field containing a hierarchic data in a list box set to this mode, you get a full preview structure, where you can expand collapse and make selections in either nodes or leaves. This functionality is the logical extension of the hierarchy resolution script syntax initially introduced in QlikView 8.50.

How it Works

This functionality is activated on the **General** tab of the list box properties, **Show As Treeview**. Such a field can be built using the path parameter of the hierarchy prefix. The Separator sets the character that should be interpreted as the separator in the path used for the Tree view.

Note: this control only works if the field contains the path representation of nodes in the hierarchy.

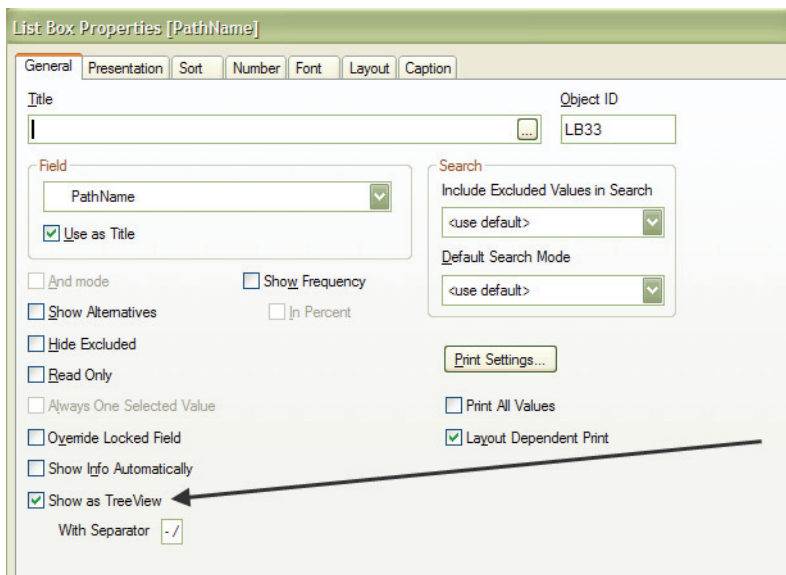


Figure 4. Setting the List Box for Tree View



Figure 5. The expanded Tree View List Box

Mini Charts

Now you can design a straight table with mini charts displayed for the expression value of a cell.

Once you have built an expression in a Straight Table you can choose to show it as a sparkline and add a dimension over which to aggregate the expression. Other display options for the mini chart include lines, dots, bars, or whiskers.

How it Works

A representation mode for straight table cells called Mini Chart displays the expression value in a miniature chart with the expression aggregated over an extra dimension field. The chart will be drawn in the available table cell, and the mini chart dimension can be defined and visual settings for the chart can be modified via the **Mini Chart Settings** button.

Regional Scorecard										XL
Region	Sales EUR Ranking 2009	2009	Sales EUR 2008	%	Sales Trends 2008 - 2009	Budget EUR 2009	Budget EUR 2009	%	Sales vs Budget 2009	
Total	21,421,961		58,279,041	37%			66,043,534	32%		
NORDIC	10,237,751		22,633,998	45%			25,394,817	40%		
USA	5,211,325		16,394,030	32%			18,905,175	28%		
JAPAN	2,390,335		7,386,439	32%			8,900,320	27%		
UK	1,925,049		7,404,419	26%			6,630,265	29%		
GERMANY	934,179		2,100,962	44%			2,759,447	34%		
SPAIN	723,323		2,359,194	31%			3,453,510	21%		

Figure 6. Straight Table with Mini Chart Example

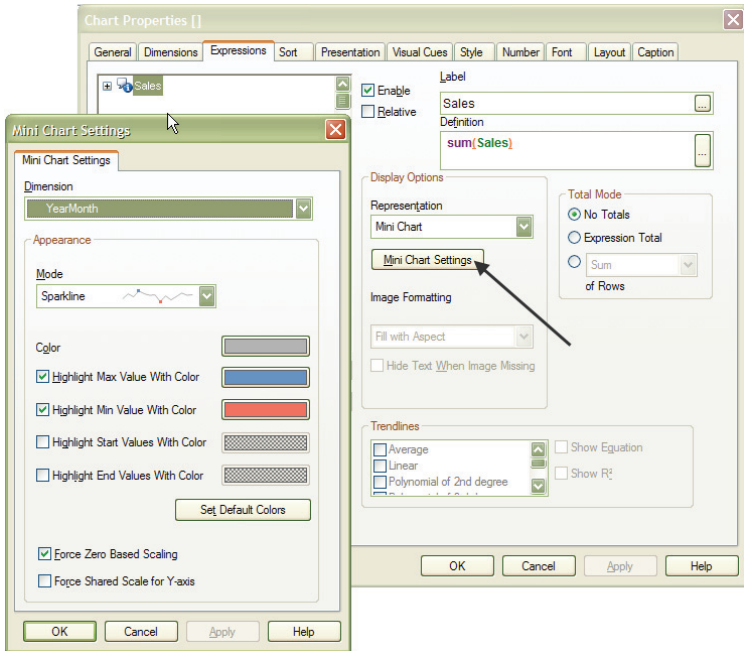


Figure 7. Mini Chart Settings

Trellis Chart

The Trellis chart fills a common need to look at the same information across the different values of a dimension. Think of it as a matrix of charts within a single QlikView Chart Object.

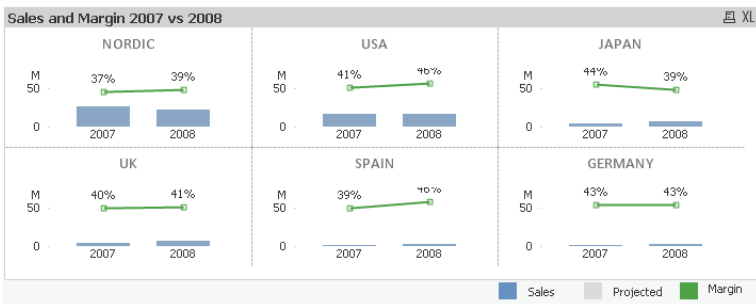


Figure 8. Trellis Chart Example

How it Works

A trellis chart uses the first dimension of a chart as the iteration dimension for producing an array of charts. A second dimension can be used as well. The trellis chart settings are available from the dimensions tab of the chart object.

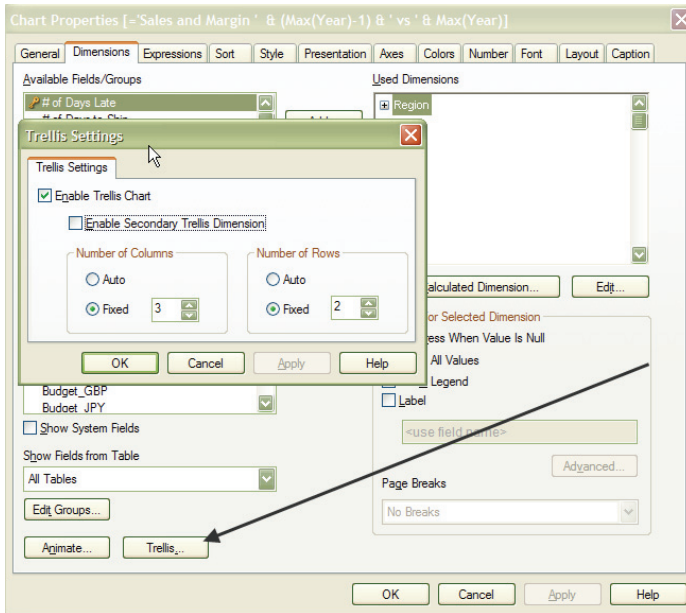
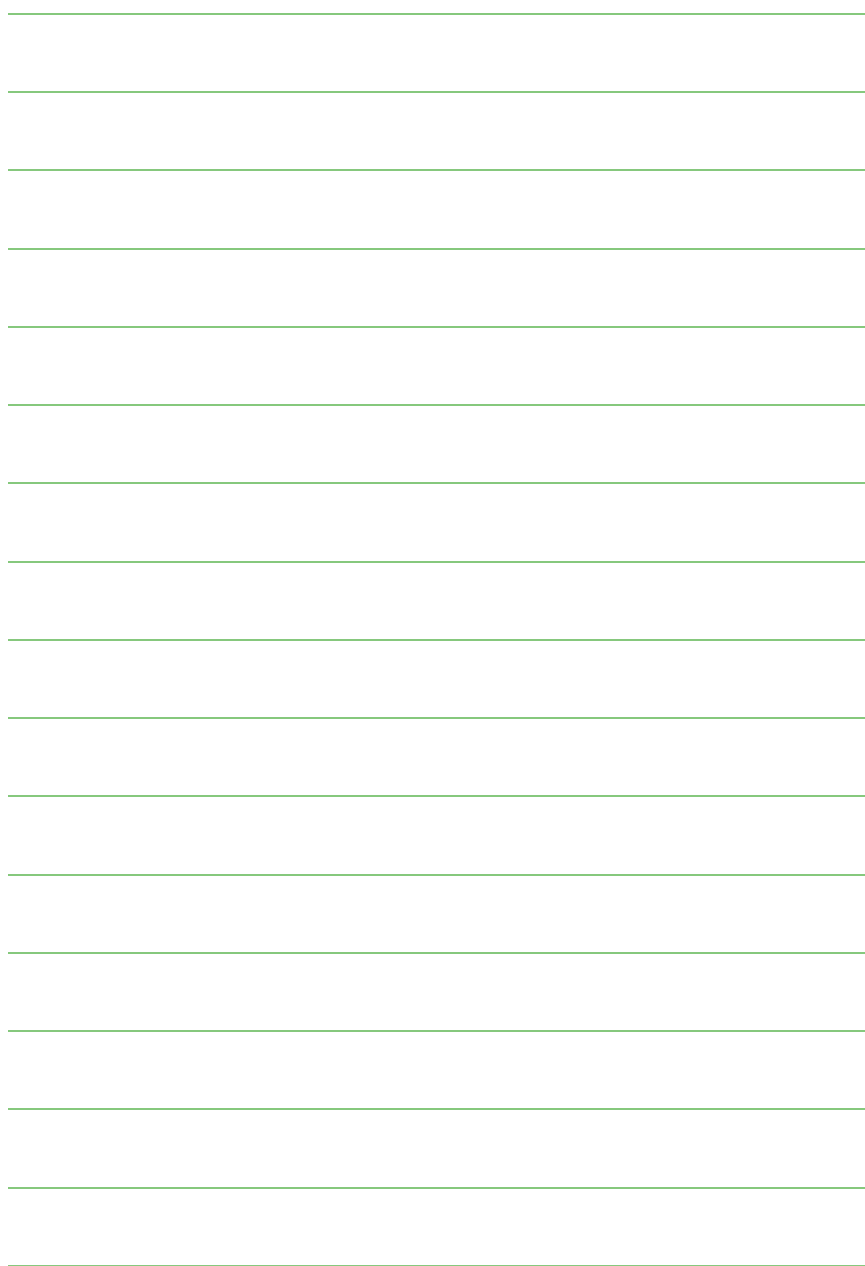


Figure 9. Trellis Chart Button and Properties, accessed from the Dimensions Tab of the chart





EXERCISE – ADVANCED CHARTS, TABLES & OBJECTS

Do: Search Object

- 1 Navigate to the c:\QlikViewTraining\DesignerII\Chapter3 directory and open the **QVDesignerII_Chapter3.qvw** file.
- 2 Save a copy of the file to preserve the original in case you want to start again from the beginning later. Do this by using the **File | Save As** command. There is also a QlikView file ending in “_Solution.qvw” containing the completed exercise for your reference.
Search objects can be used to search through one or several fields in the application. By using the search object, you can actually make selections in fields that are not present on the sheet where you have the search object.
- 3 Start by adding the two list boxes *Country* and *SupplierCountry*
- 4 Right click on the sheet and open **New Sheet Object** and select **Search Object**.
- 5 Select the radio button **Selected fields**
- 6 Select the fields *Country* and *Suppliers.Country*.
- 7 Go to the Presentation page and select the check boxes for **Include Excluded** and **Highlight Substring**.
The Presentation page of the Search object properties
 - **Include Excluded** - Will include values in the search that has been excluded by previous selections.
 - **Highlight substring** - Will highlight in yellow what you have typed so far in values still possible.
- 8 Click **OK** to close the Properties dialog.
- 9 Click on the Search object to open it for searching.
- 10 Start typing **S**.
As you can see there will be a box with the search in both fields.
If there are more values than the box is allowed to show, only the number of matches will be seen in the result window.
- 11 Continue typing **w** in order to search for *Sweden*.
As you can see, the number of matches can now be seen in the Search result box and the typed values are highlighted in yellow.
- 12 Type the letter **e** and press Enter when only *Sweden* is available in both list boxes.
Sweden is now selected in both list boxes.
- 13 Add another list box, *ProductName* to the sheet.
The products now available are products that has been sold to customers in *Sweden*, supplied by *Swedish suppliers*.



Tip: If you want to find different countries in the two list boxes, search for one country first, and click on the value in the list box where you want to make the selection. Then search for the other country and click on that value in the other list box.

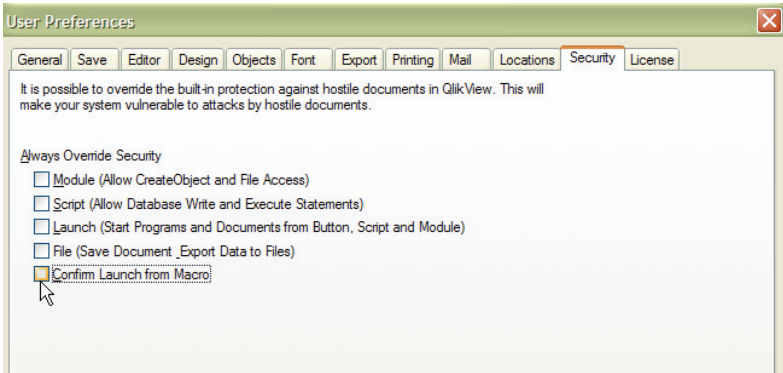
Extra Credit

- 1 Find out what products have been sold to *Spain* by *Canadian suppliers*.
- 2 Now create a Search box that searches through all fields of the application.

Do: Clickable Links

- 1 Continue working in your copy of the QVDesignerII_Chapter3.qvw file. Where *Name* and *Link* would be fields loaded into QlikView.
- 2 Right click on the Sheet and select **New Sheet Object - Chart**.
- 3 Select **Straight table** on the General page and click next.
- 4 Select *City* as Dimension and click **Next**.
- 5 Type the following text into the Expression window:
 ='City web page<URL>'&CityURL
 The city's web page will be shown in each cell. CityURL is a field that holds webpages containing information on the different cities.
- 6 Click **OK**.
- 7 Make sure to Select **No Totals** on the *Expression* tab.
- 8 Select **Link** in the Representation Drop down in order to get a clickable link.
- 9 Click **OK** to close the Chart Properties dialog.
- 10 Click on a City web page link to open the webpage for that city.

Note: When clicking on the link, you might be prompted to “Approve Application for Launch” before the link will open in your web browser. Should you wish to launch the links automatically (e.g. without this prompt for approval), de-select the **Settings | User Preferences | Security** check-box for **Confirm Launch from Macro**



Do: Tree View List Boxes

If you have a field read in as a hierarchy field or with different sub-strings, you can use a list box in Tree View. For now we are going to look at a field created by a hierarchy table. The field *PathName* can be used in a tree list box.

- 1 Continue working in your copy of the *QVDesignerII_Chapter3.qvw* file.
- 2 Right click on the sheet and choose **Select Fields**.
- 3 Select the field *PathName* and click **OK**.
As you can see, there are different numbers of sub strings in this list box. The substrings are divided with "- / -". The Americas and Europe values are parent nodes to the other values. When you have a structure similar to this, you can use a Tree list box to get a better overview.
- 4 Right click on the *PathName* list box and go to the Properties of the list box.
- 5 On the General page of the Properties dialog, check **Show as TreeView**.
- 6 Type in "- / -" as separator [space "-" space "/" space "-" space] and click **OK**.

When shown in TreeView, the field values will be divided into the sub-strings of the value Shown as a tree with the Parents on top and a plus to open up the sub levels.

Tip: This can provide a clearer view on fields in a hierarchy.



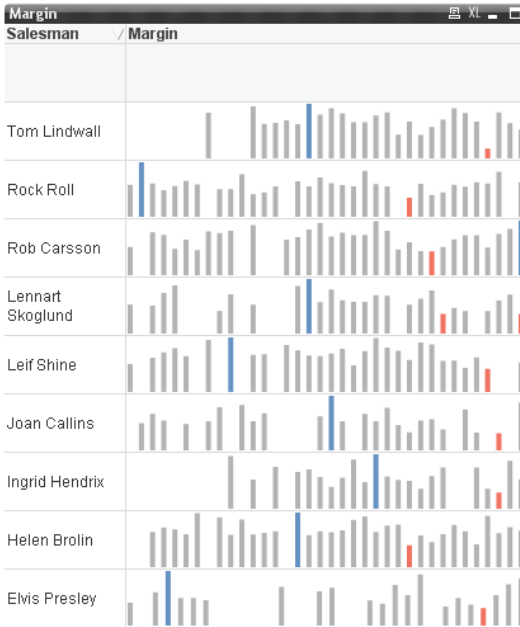
Do: Mini Charts

Sparklines are a type of mini chart that show an expression aggregated over an extra dimension field. When you have built an expression you can choose to show it as a sparkline and add a dimension over which to aggregate the expression. We are going to look at how our sales person has performed over time.

- 1 Continue working in your copy of the *QVDesignerII_Chapter3.qvw* file.
- 2 Create a Straight table by right clicking on the sheet and select **New Sheet Object** and **Chart**.
- 3 On the General page of the chart properties, select the **Straight table** chart and click **Next**.
- 4 Select *Salesman* as the Dimension
- 5 Go to the Expressions page and enter the expression `sum(Sales)` and click **OK**.
- 6 Label the expression *Sales*.
- 7 In the middle of the Expression page, select **Mini Chart** in the **Representation** Drop down.
- 8 Select the field *YearMonth* in the **Dimension** Dropdown.
- 9 Select **Sparkline** in the **Mode Dropdown** and let the colors be as default.
- 10 Continue to the Presentation page and check the **Wrap cell text** radio button. Set the number of rows to 3.
- 11 Click **OK** to close the chart dialog.
- 12 A chart with a sparkline showing Sales over time is now visible for each Salesman.
- 13 Create a Straight table with *Salesman* as the Dimension.
- 14 Create an expression showing Margin % as $((\text{sum}(\text{Sales}) - \text{sum}(\text{COS})) / \text{sum}(\text{Sales}))$
- 15 Show the expression as a **Sparkline - Bar** with *YearMonth* as the Dimension.



16 Make sure to wrap the cell text and show three lines in the straight table.



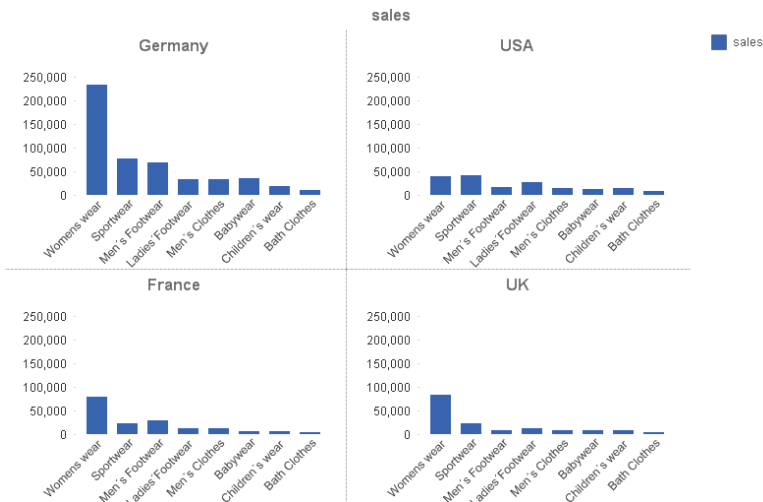
Do: Trellis Chart

The Trellis chart fills a common need to look at the same information across the different values of a dimension. Think of it as a matrix of charts within a single QlikView Chart Object.

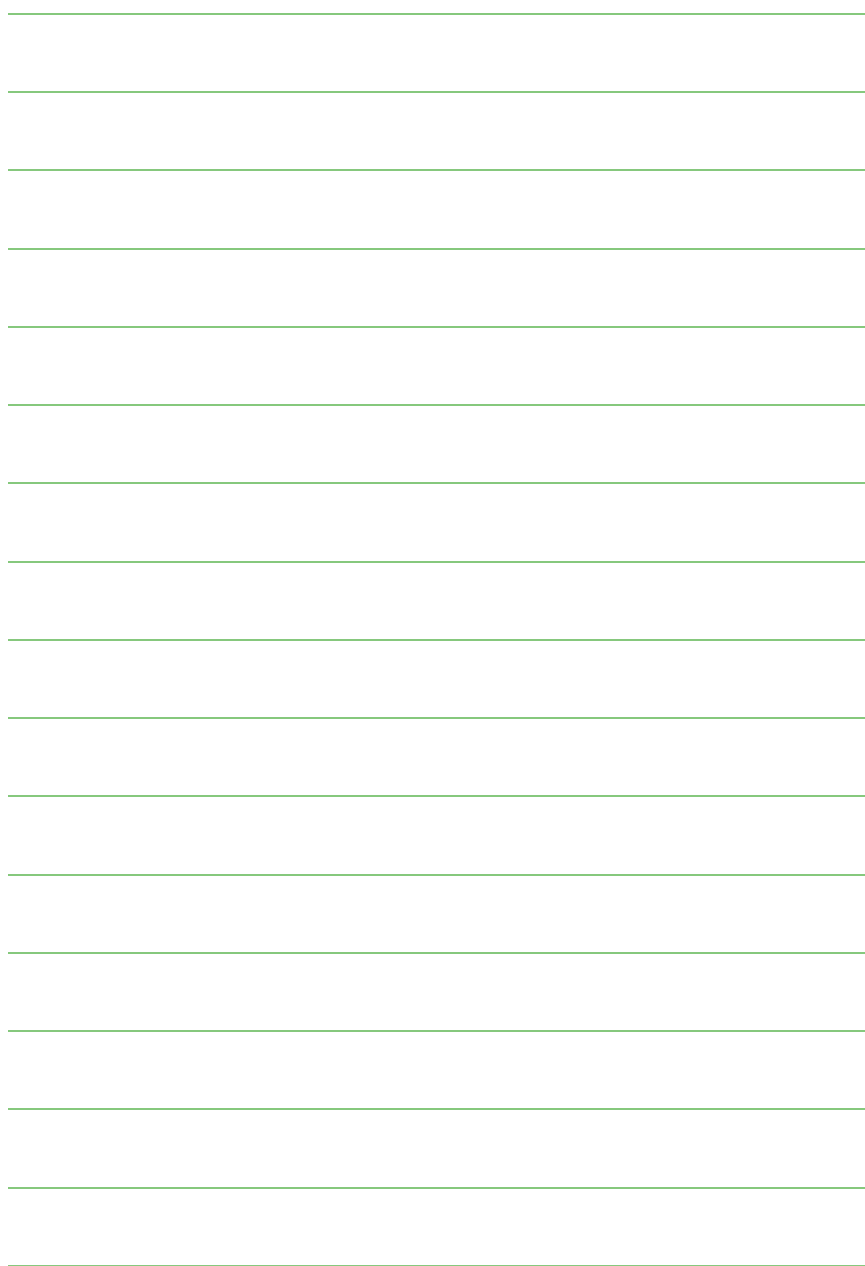
- 1 Continue working in your copy of the **QVDesignerII_Chapter3.qvw** file.
- 2 Right click on the sheet and select **New sheet object - Chart**.
- 3 Select **Bar chart** in the General page of the chart dialog.
- 4 Click **Next** and select the dimensions *Country* and *CategoryName* in the Dimensions page. Make sure to uncheck **Label** for the *Country* dimension.
- 5 In the bottom of the same page there is a button, **Trellis...** Press this button to enter the Trellis Settings.
- 6 Make sure that **Enable Trellis chart** is checked and that **Enable Secondary Trellis Dimension** is unchecked.
- 7 Set the number of columns to **Fixed** with two columns and do the same with number of rows.



- 8 Click **OK**.
- 9 Go to the Expressions page and add the expression sum(Sales)
- 10 Give the expression the label **Sales**.
- 11 Go to the Sort page and sort the *Country* dimension by **Y-value** and sort *CategoryName* by **expression** sum(Sales) Descending.
- 12 Continue to the Axes page and set the **Primary dimension label** to diagonal.
- 13 Click **OK** to close the Chart dialog.
- 14 Now create a **Line chart** that enables Trellis mode over the dimension *Salesman*.
- 15 The subcharts should have *YearMonth* as Dimension and sum(Sales) as Expression.
- 16 Label the expression *Sales*.
- 17 Click **OK** to finish the chart.
- 18 Look at the chart and then decide what number of columns and rows to use in order to get a good presentation of the chart.
- 19 Go to the presentation page and limit the number of values shown to an appropriate number and make sure to use an X-axis scrollbar.







4 ADVANCED CALCULATIONS IN SHEET OBJECTS

Objectives

- Introduce advanced calculations in charts and tables, including:
 - Set Analysis
 - Dollar-Sign Expansion
 - AGGR Function
- Complete exercises using examples of each of these functions

Set Analysis

QlikView has always been good at calculating aggregates for the current selection of data. However, when you wanted to compare results for different selections in the same chart, you needed to either prepare data in the script or resort to rather complicated expressions with if clauses.

Set analysis changes all that, by making it possible to modify any aggregation function with an arbitrary selection set.

The set may be defined as a bookmark, as an on-the-fly selection in one or more fields, as a function of current selections, the inverse of current selections, previous selections, all data, etc., etc.

The possibilities are endless and yet the syntax is fairly simple and straightforward.

Indirect Set Analysis

Prior to version 9.00, set analysis in QlikView 8.50 was restricted to stating direct selections in a field. With the new extensions in QlikView 9.00 selections in a field can be stated based on selections in another field, such as

Select all possible values in *Customers* based on *Sales* last year.

Overview

Sets can be used in aggregation functions. Aggregation functions normally aggregate over the set of possible records defined by the current selection. But an alternative set of records can be defined by a set expression. Hence, a set is conceptually similar to a selection.

Note: A set expression is always enclosed in curly brackets when used, e.g.{BM01}.

Set Identifiers

There are two constants that can be used to denote record sets. They are **0** and **1**. They represent an empty set and a full set of all the records in the application, respectively.

The **\$** sign represents the records of the current selection. The set expression **{ \$ }** is, therefore, the equivalent of not stating a set expression at all. **{ 1-\$ }** is all the more interesting as it defines the inverse of the current selection, that is, everything that the current selection excludes.

Selections from the Back/Forward stack can be used as set identifiers, by use of the dollar symbol: **\$1** represents the previous selection and is equivalent to pressing the Back button. Similarly, **\$_1** represents one step forward and is equivalent to pressing the Forward button. Any unsigned integer can be used in the Back and Forward notations. **\$0** represents the current selection.

Finally, bookmarks can be used as set identifiers. Either the bookmark ID or the bookmark name can be used, **BM01** or **MyBookmark**.

Set Operators

Several operators are used in set expressions. All set operators use sets as operands, as described above, and return a set as result. The operators are as follows:

- +** **Union.** This binary operation returns a set consisting of the records that belong to any of the two set operands.
- **Exclusion.** This binary operation returns a set of the records that belong to the first but not the other of the two set operands. Also, when used as a unary operator, it returns the complement set.
- *** **Intersection.** This binary operation returns a set consisting of the records that belong to both of the two set operands.
- /** **Symmetric difference (XOR).** This binary operation returns a set consisting of the records that belong to either, but not both of the two set operands.

The order of precedence is

- 1 Unary minus (complement)
- 2 Intersection and Symmetric difference
- 3 Union and Exclusion.

Within a group, the expression is evaluated left to right. Alternative orders can be defined by standard brackets, which may be necessary since the set operators do not commute, i.e. $A + (B - C)$ is different from $(A + B) - C$ which in turn is different from $(A - C) + B$.

Set Operator Examples:

sum({1-\$} Sales)

returns the sales for everything excluded by the current selection.

sum({\$*BM01} Sales)

returns the sales for the intersection between the current selection and bookmark BM01.

sum({-(\$+BM01)} Sales)

returns the sales excluded by current selection and bookmark BM01.

Note: The use of set operators in combination with basic aggregation expressions involving fields from *multiple* QlikView tables may cause unpredictable results and should be avoided. E.g. if “Quantity” and “Price” are fields from different tables, then the expression `sum({$*BM01} Quantity * Price)` should be avoided.

Set Modifiers

A set can be modified by making an additional or a changed selection.

Such a modification can be written in the set expression.

The modifier consists of one or several field names, each followed by a selection that should be made on the field, all enclosed by `<` and `>` as in

`<Year={2007, 2008}, Region={US}>`

Field names and field values can be quoted as usual, e.g. `<[Sales Region]=('West coast', 'South America')>`.

There are several ways to define the selection:

A simple case is a selection based on the selected values of another field, e.g. `<OrderDate = DeliveryDate>`. This modifier will take the selected values from “DeliveryDate” and apply those as a selection on “OrderDate”.

Note: If there are many distinct values – more than a couple of hundred – avoid this operation because it is CPU intensive.

The most common case, however, is a selection based on a field value list enclosed in curly brackets, the values separated by commas, e.g. `<Year = {2007, 2008}>`. The curly brackets here define an element set, where the elements can be either field values or searches of field values.

A search is always defined by the use of double quotes, e.g. `<Ingredient = {"*Garlic*"}>` will select all ingredients including the string 'garlic'.

Note: Searches are case-insensitive and are made over excluded values too.

Tip: Empty element sets, either explicitly e.g. `<Product = {}>` or implicitly e.g. `<Product = {"Perpetuum Mobile"}>` (a search with no hits) mean *no product*, i.e. it will result in a set of records that are *not* associated with any product.

Further, the selection within a field can be defined using set operators and several element sets, such as with modifier

`<Year = {"20*", 1997} - {2000}>`

which will select all years beginning with “20” in addition to “1997”, *except* for “2000”.

The above notation defines new selections, disregarding the current selection in the field. However, if you want to base your selection on the current selection in the field and add field values, e.g. you may want a modifier `<Year = Year + {2007, 2008}>`. A short and equivalent way to write this is `<Year += {2007, 2008}>`, i.e. the assignment operator implicitly defines a union.

Also implicit intersections, exclusions and symmetric differences can be defined using “*=”, “-=” and “/=”.

Finally, for fields in and-mode, there is also the possibility of forced exclusion. If you want to force exclusion of specific field values, you will need to use “~” in front of the field name.

Note: A set modifier can be used on a set identifier or on its own. It cannot be used on a set expression. When used on a set identifier, the modifier must be written immediately after the set identifier, e.g. `{<Year = {2007, 2008}>}`. When used on its own, it is interpreted as a modification of the current selection.

Dollar-Sign Expansion

Dollar-sign expansions are definitions of text replacements used in the script or in expressions. This process is known as expansion - even if the new text is shorter. The replacement is made just before the script statement or the expression is evaluated. Technically, it is a macro expansion.

A macro expansion always begins with **\$**(and ends with) and the content between brackets defines how the text replacement will be done. To avoid confusion with *script* macros we will henceforth refer to macro expansions as dollar-sign expansions.

Note: Macro expansion is unrelated to script macros (VB or Java script defined in the script module).

Dollar-sign Expansion using a variable

When using a variable for text replacement in the script or in an expression, the syntax

\$ (variablename)

is used. **\$(variablename)** expands to the value in **variablename**. If **variablename** does not exist the expansion will be the empty string.

For numeric variable expansions, the syntax

\$ (variablename)

is used. **\$(variablename)** always yields a legal decimal-point reflection of the numeric value of **variablename**, possibly with exponential notation (for very large/small numbers). If **variablename** does not exist or does not contain a numeric value, it will be expanded to 0 instead.

Dollar-Sign Expansion with Parameters

Parameters can be used in variable expansions. The variable must then contain formal parameters, such as **\$1**, **\$2**, **\$3** etc. When expanding the variable, the parameters should be stated in a comma separated list.

If the number of formal parameters exceeds the number of actual parameters, only the formal parameters corresponding to actual parameters will be expanded. If the number of actual parameters exceeds the number of formal parameters, the superfluous actual parameters will be ignored.

The parameter **\$0** returns the number of parameters actually passed by a call.

Dollar-Sign Expansion with an Expression

Expressions can be used in dollar-sign expansions. The content between the brackets must then start with an equal sign:

`$(=expression)`

The expression will be evaluated and the value will be used in the expansion.

Example:

`$(=Year(Today()))` returns the calendar year based on the system date, so if your system date is 28 May 2009, **2009** would be returned

`$(=Only(Year)-1)` returns the year before the selected one

AGGR Function

AGGR is a powerful QlikView function that returns a set of values of expression calculated over dimensions. The result can be compared to the expression column of a local chart, evaluated in the context where the **aggr** function resides. Each dimension must be a single field and cannot be an expression (e.g. a calculated dimension).

If the expression argument is preceded by the *nodistinct* qualifier, each combination of dimension values may generate more than one return value, depending on underlying data structure. If the expression argument is preceded by the *distinct* qualifier, or if no qualifier is used at all, each combination of dimension values will generate only one return value.

By default, the **aggr** function will aggregate over the set of possible records defined by the selection. An alternative set of records can be defined by a **set expression**.

By using this function in calculated dimensions it is possible to achieve nested chart aggregation in multiple levels.

When **aggr** is used in chart expressions it is possible to achieve *sum-of-rows* totals in a pivot table.

Examples:

```
aggr( sum(Sales), Country )
aggr( nodistinct sum(Sales), Country )
aggr( sum(Sales), Country, Region )
count( aggr( sum(Sales), Country ))
```



Advanced Calculations Exercises

Do: Set Analysis Exercise

Create a Straight Table chart that displays a comparison of annual sales by CompanyName based on the year selected by the user.

- 1 Navigate to the c:\QlikViewTraining\DesignerIIL\Chapter4 directory and open the **QVDesignerII_Chapter4.qvw** file.
- 2 Save a copy of the file to preserve the original in case you want to start again from the beginning later. Do this by using the **File | Save As** command. There is also a QlikView file ending in “**_Solution.qvw**” containing the completed exercise for your reference.
- 3 Navigate to the **Workspace** Sheet.
- 4 Double check to be sure there is a list box on the sheet for the *Year*. If not, add one.
- 5 Right-click in a blank area of the sheet and **New Sheet Object | Chart** from the context menu.
- 6 Click on the **Straight Table** icon (the lower right corner of the Chart Types) and type **Annual Comparison** in the Window Title. Click on the **Next** button.
- 7 Add **Customer** to the **Used Dimensions** and click **Next**
- 8 Create the following three **Expressions** using the **Labels** provided:
 1. Label
=Only(Year)
 1. Expression
Sum({\$<Year={\$(=Only(Year))}>} Sales)
 2. Label
=Only(Year)-1
 2. Expression
Sum({\$<Year={\$(=Only(Year)-1)}>} Sales)
 3. Label
=Only(Year) & ' vs ' & (Only(Year)-1)
 3. Expression
Sum({\$<Year={\$(=Only(Year))}>} Sales) - Sum({\$<Year={\$(=Only(Year)-1)}>} Sales)
- 9 Click **Finish**
- 10 **Save** your QlikView file and then continue to edit the Annual Comparison straight table.



11 Set the Sort order to match the depiction, below, remembering that Customer should be set to Text.

Chart Properties [Annual Comparison]

General Dimensions Expressions **Sort** Presentation Visual Cues Style Number Font Layout Caption

Columns

Priority

- =Only(Year) & ' vs ' & (Only(Year)-1)
- Customer
- =Only(Year)
- =Only(Year)-1

Promote Demote

Sort by

- ☐ Expression
- ☐ Frequency
- ☒ Numeric Value Ascending
- ☐ Text
- ☐ Load Order

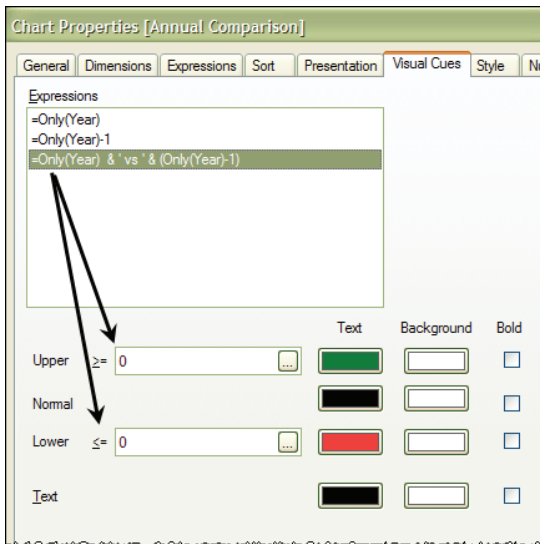
☐ Override Group Sort Order

☒ Allow Interactive Sort

OK Cancel Apply Help



- 12 On the **Visual Cues** tab, make the negative values for the year-to-year comparison red and the positive values green.



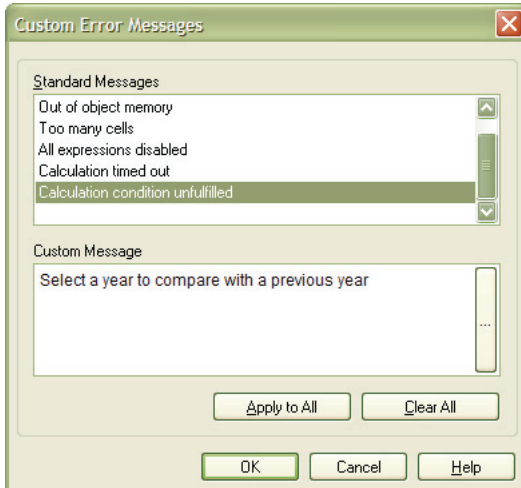
- 13 Return to the **General** tab and add a **Calculation Condition** to ensure that the user selects a Year to begin the comparison by entering the following into the **Calculation Condition** box

Count(distinct [Year])=1

- 14 Click on the **Error Messages** button on the General tab and then on **Calculation Condition Unfulfilled** in the **Standard Messages** list.



- 15 Type: **Select a Year to compare with a previous year** in the Custom Message box and click **OK**.



- 16 Click **OK** again to close the chart properties dialog.
 17 With **2009** selected in the Year list box you added at the beginning of the Exercise, your straight table should look something like the one below:

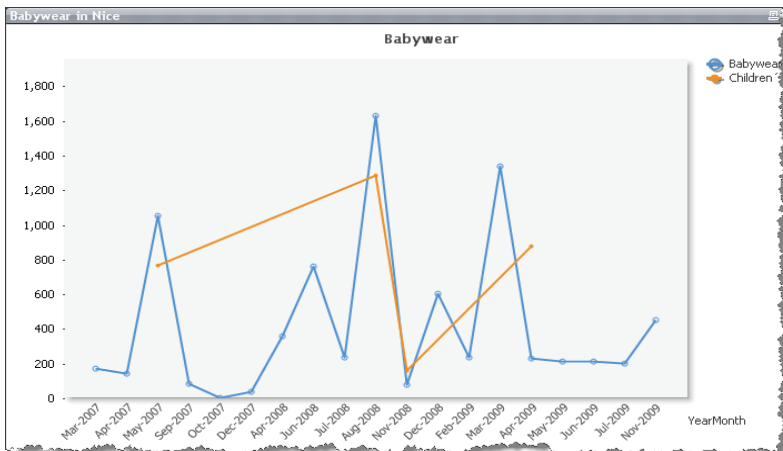
Annual Comparison			
Customer	2009	2008	2009 vs 2008
Total	772,576	600,569	+172,006
Warp AG	10,615	38,959	-28,344
Eintrach GS	23,757	48,611	-24,854
Boombastic	13,578	29,615	-16,037
Paintho da Gama	435	14,378	-13,942
Davenport Fashion	1,845	13,998	-12,153
Don Balón	25,184	34,922	-9,738
Th Fashing	48,181	57,507	-9,325
The Corner Store	28,029	36,610	-8,581
Cloe do Pau	146	5,092	-4,945
Too Hot 4U	848	5,628	-4,780
Stephanies	910	5,618	-4,708
La Ropa Vieja	1,407	4,862	-3,456
Autokleider	566	3,347	-2,781
Bobby Socks	0	2,458	-2,458
Casual Clothing	5,006	7,341	-2,334
De la Vita	1,532	3,380	-1,847
Fast Sunglasses	0	1,700	-1,700

- 18 Save your work.



Do: Advanced Set Analysis

- 1 Continue working in the file you have been using in this exercise chapter so far.
- 2 Create a chart that compares sales of products in the category *Babywear* with products in the category *Children's Wear* over time for the *Nice* sales office.
- 3 To do this you will need to create an expression using Set analysis and **\$ Expansion** instead of traditional **if()** statements.
- 4 Set the category name to find *Babywear* and the office to be 4.
- 5 Pay attention to the **<>** and **{}**.



Solution:

```
sum({$<CategoryName={'Babywear'}, Office={4}>} Sales)
sum({$<CategoryName={'Children's wear'}, Office={4}>} Sales)
```

Do: AGGR Function Exercise

- 1 Continue working in the file you have been using in this exercise chapter so far.
- 2 Create a table that shows if there is any link between the number of orders placed by customers and the average order value. The table should provide information on how many customers have placed one order, two orders etc, and also the average order value.



- 3 There are three steps to this process. First, create a calculated dimension for the number of orders (as in “how many customers had one order, two orders, three orders, etc.). This requires **aggr**. First, to count the number of orders, use

`count(distinct OrderID)`

and then to aggregate those against the *Customer* Dimension use

`aggr(....., CustomerID)`

Putting it together,

`aggr(count(Distinct OrderID), CustomerID)`

will create the necessary order “buckets” based on customer

- 4 Next you need a count of customers to populate the # of order “buckets” we created in the first step.

`Count(distinct CustomerID)`

- 5 And, finally, we need to find the average order amount.

Sales is from the Sales Detail Table and is a line item for every product sold. Thus one order could have several products and thus several lines with Sales data, so we need to aggregate by OrderID to get the total sales amount for an order

`aggr(sum(Sales),OrderID)`

gets you that number and adding the **avg()** gives you the requested average order amount

`avg(aggr(sum(Sales),OrderID))`

# Orders			
# of Orders	# of Customers	Avg Order Amount	
	92	\$2,149.17	
0	2	-	
1	13	\$760.91	
2	7	\$1,646.10	
3	8	\$1,377.66	
4	9	\$951.05	
5	10	\$1,158.89	
6	10	\$1,367.98	
7	3	\$1,177.01	

5 GAUGES

Objectives

- Define what we mean by a dimensionless expression
- Understand when to use a Gauge
- Practice advanced expressions with gauge exercises

Gauges are most commonly used in QlikView dashboards. They represent a dimensionless expression, and reflect the selection status (what has been clicked) in the QlikView application.

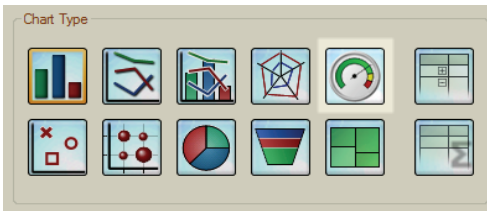
This chapter goes over best practices for creating meaningful gauges in QlikView.

Why Use a Gauge Chart?

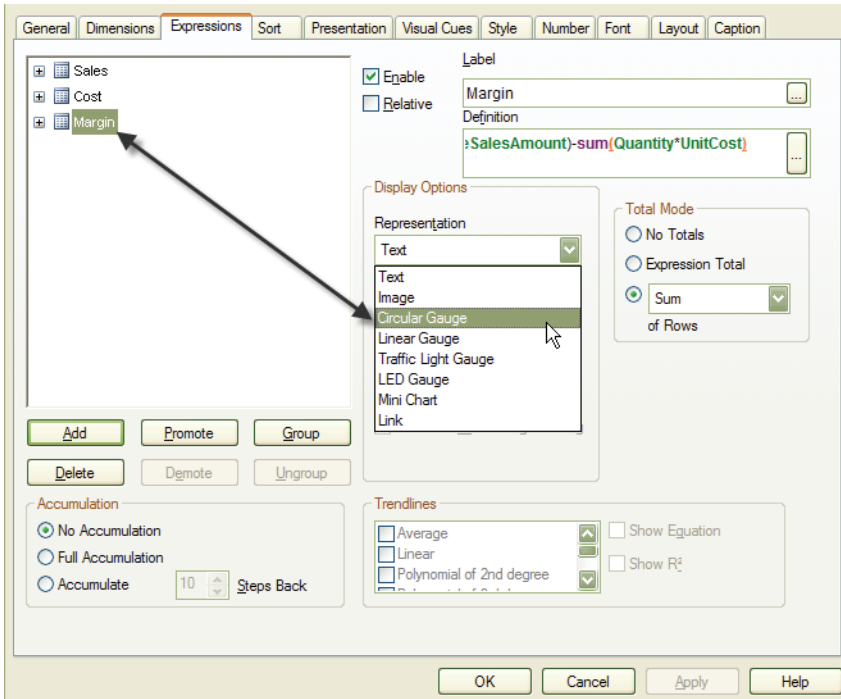
The Gauge is, first of all, a Chart Type in QlikView. It is used for displaying the result of a single dimensionless expression, like a Key Performance Indicator (KPI). Gauges are flexible and can encompass many different display options.

Note: Gauges can be associated with Actions to trigger different events. Clicking on a gauge might open a pivot table of supporting information, for example.

You create a gauge like any other chart, starting out with a new sheet object, chart, and then clicking on the gauge icon in the chart type selection area, as below.



Expression values in the data cells of a straight or pivot table can be shown as gauges. This is accomplished on the **Expressions** page of the straight table or pivot table properties.



Selecting the gauge as the expression display option will change the depiction in the **Expressions** page of the **Chart Properties** dialog and will activate the **Gauge Settings** button, below.

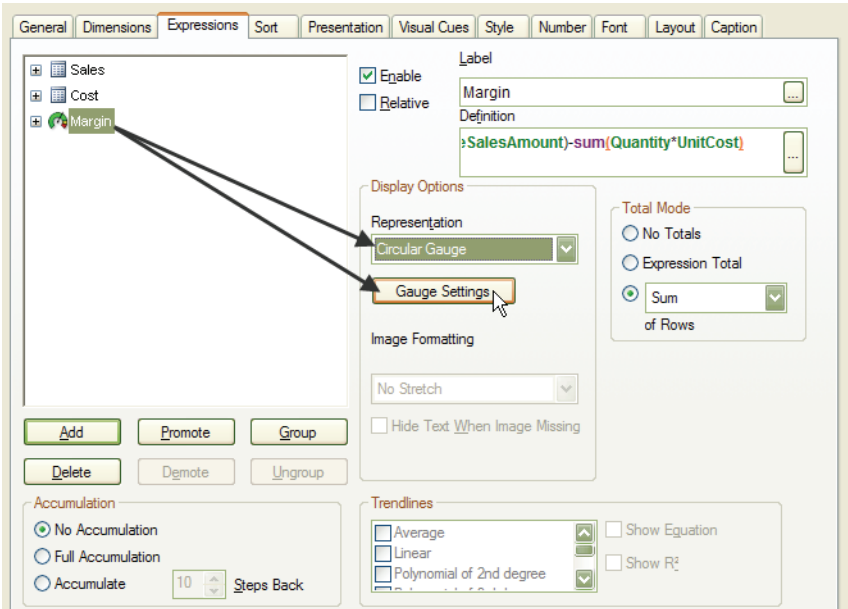


Figure 10. The addition of the Gauge Display Options setting

Gauges work best when you want to show a single number in a specific context, based upon whatever the user has clicked.

Types of Gauges

The following table displays the same gauge in six different formats.

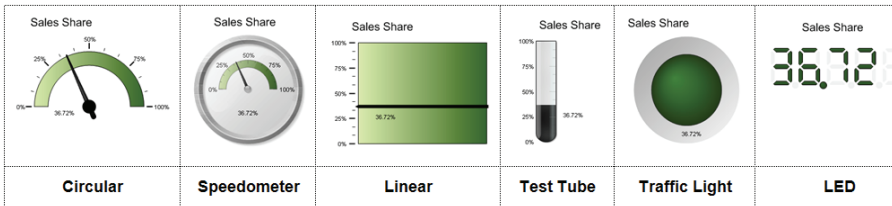
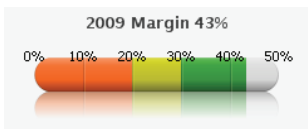


Figure 11. Gauge examples



QlikView designers should experiment with the different gauge types to determine the best choice for the expression they are evaluating.

Building Gauges

Say you are interested in determining the percentage of sales for a given selection status. You could represent this in a gauge, using the expression

Sum (LineSalesAmount)/Sum (All LineSalesAmount)

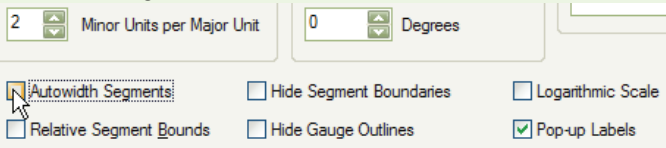
You would start out by —

- 1 Creating a **New Sheet Object: Chart**, and selecting the **Gauge Chart** icon in the **Chart Type** group.

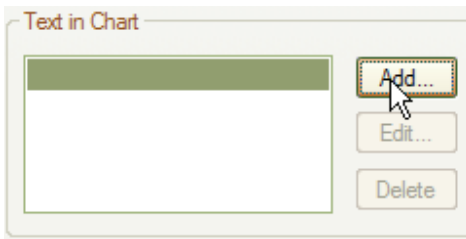
Note: you could use the **Quick Chart Wizard** from the **Layout** menu, as well. The advantage of using the wizard is that you will avoid the confusion of being asked to select a dimension for the (dimensionless) gauge. The disadvantage is that only a limited number of gauge styles are presented in the **Quick Chart Wizard**. However, you can go back at any point to change settings in the gauge chart that you create. Right-click the gauge and select its **Properties**.

- 2 Realizing that the gauge chart is dimensionless, you would ignore the **Dimensions** page and proceed directly to the **Expressions** page and enter the expression:
`Sum (LineSalesAmount)/Sum (All LineSalesAmount)`
- 3 Label the expression **% of Sales**
- 4 Advance to the **Style** sheet and select an appropriate gauge *Look*.
- 5 Next, make adjustments on the **Presentation** page of the (gauge) **Chart Properties** and notice that the options change depending on the type of gauge you are creating.

Tip: you will need to de-select the **Autowidth Segments** box to add or remove segments or change the **Min/Max**, the **Labels** and **Bounds**.

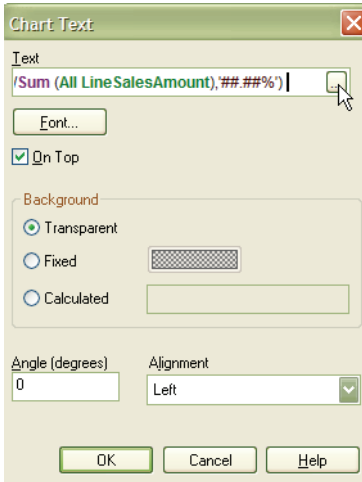


- 6 Add a **Text in Chart** expression duplicating the expression upon which the gauge is based. That way your users will see the value of their selection represented both in the gauge and in a numeric value.



- 7 Type the following into the text box:
`=num(Sum (LineSalesAmount)/Sum (All LineSalesAmount), '##.##%')`

or open the **Edit Expression** dialog and use the paste functions to build the expression and click **OK**. Bypass the **Colors** page.



- 8 On the **Number** page of the **Gauge Chart Properties**, click on **Fixed to 0 Decimals** and enter a **Format Pattern** of **#,##0%**
- 9 Check the **Show in Percent (%)** checkbox.
- 10 On the **Layout** page, turn off **Borders** and on the **Captions** page turn off **Captions** and click **OK** to finish.

Remember, you can manipulate the placement of the **Text in Chart** by hovering over the gauge, pressing and holding the CTRL and SHIFT keys to activate the objects within the gauge (such as the **Title** and the **Text in Chart** value).

This is a simple gauge. You can refer to the sample applications in your **Getting Started** menu for more ideas and samples.

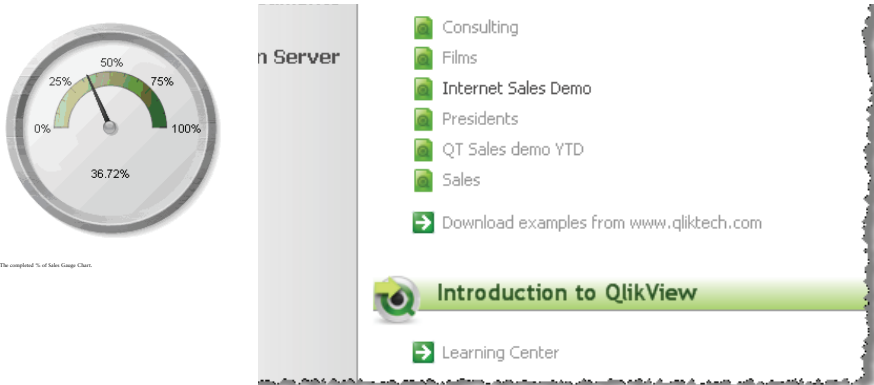


Figure 12. Getting Started - Examples List



EXERCISE

Now that you have had a chance to learn a little bit about gauges, it is time to put that knowledge to use.

Do:

- 1 Navigate to the c:\QlikViewTraining\DesignerII\Chapter5 directory and open the **QVDesignerII_Chapter5.qvw** file.
- 2 Save a copy of the file to preserve the original in case you want to start again from the beginning later. Do this by using the **File | Save As** command. There is also a QlikView file ending in “**_Solution.qvw**” containing the completed exercise for your reference.
- 3 Navigate to the **Workspace** Sheet.
- 4 One of your company’s key performance indicators is Margin Percent. Using the formula below, create a gauge chart to display this value and highlight a desired range by adding segments to the gauge.
Margin Percent is calculated based on the Margin Amount ÷ Selling Price. In our data model, this is expressed as:

$$\frac{(\text{sum}(\text{LineSalesAmount}) - \text{sum}(\text{CostOfGoodsSold}))}{\text{sum}(\text{LineSalesAmount})}$$
- 5 Customize the gauge to include a **Text in Chart** label reflecting the Margin Percent as a formatted number. An example is reproduced, below.

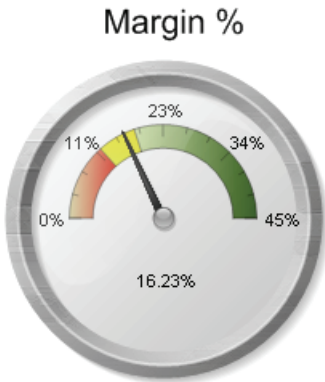
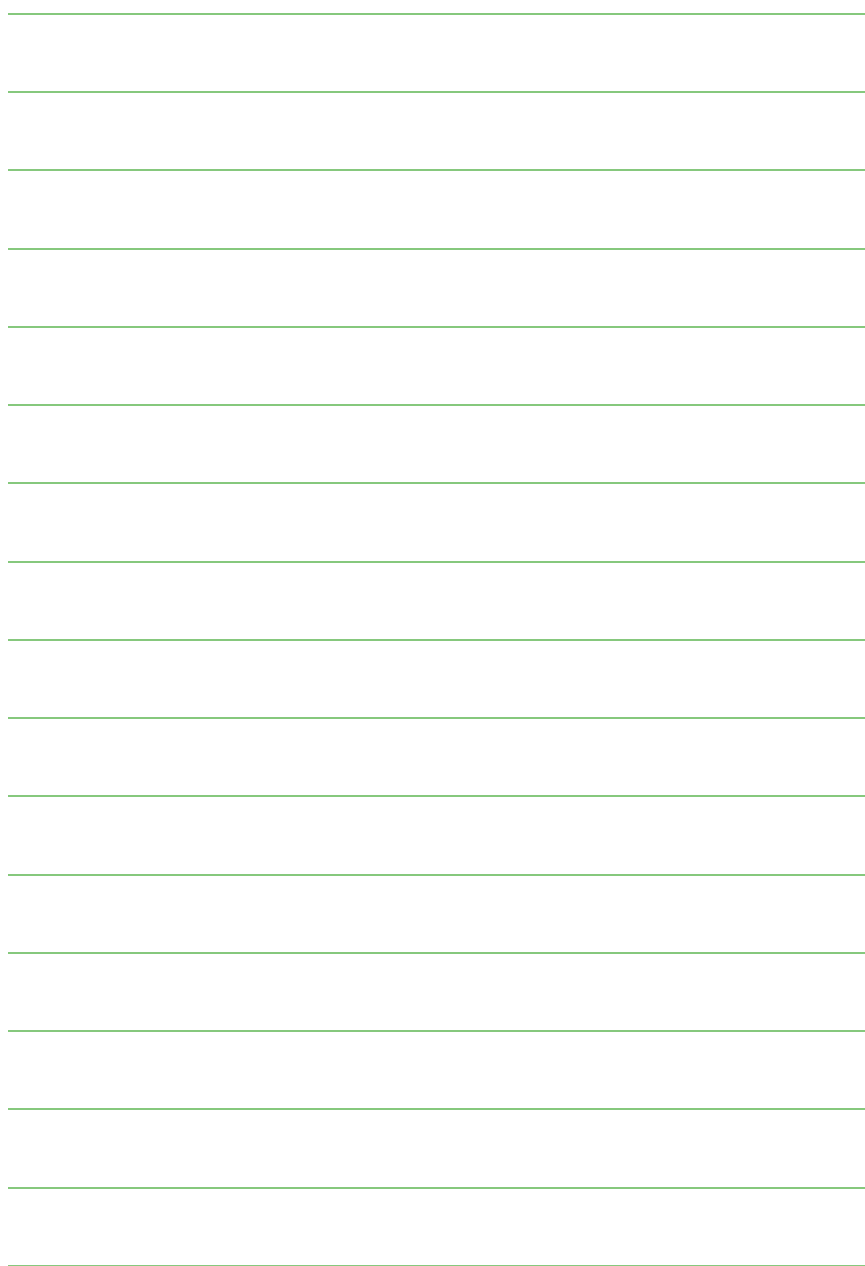


Figure 13. How your completed gauge chart for sales margin might look.



6 BUTTONS AND ACTIONS

Objectives

- Understand the use of buttons to enhance the user experience
- Learn about the common types of Actions
- Build a working button in the chapter exercise

Buttons in QlikView

Buttons can be used to execute various commands in QlikView. This is done by connecting an Action to the Button. Actions are derived from the old button shortcuts, which they also replace. Apart from offering a much wider range of operations than the old shortcuts (including most common operations on sheets, sheet objects, fields and variables), you may also define a series of operations within a single action.

The introduction of actions should greatly reduce the need for macros, which is good since macros are never efficient from a performance point-of-view.

Note: Actions are not limited to Buttons but can also be used with text objects, line/arrow objects and gauge charts can be given Actions, which are executed when clicking on the sheet object in question.

The sample QVW file contains some sample buttons located in the *Other Objects* tab. Each of the other tabs contains buttons under the QlikView logo to move between tabs and clear selections.

Creating and Formatting a Button

To create a button, choose **New Sheet Object – Button** from the **Layout** menu to open the **Button Properties** dialog.

On the **General** page you format the button.

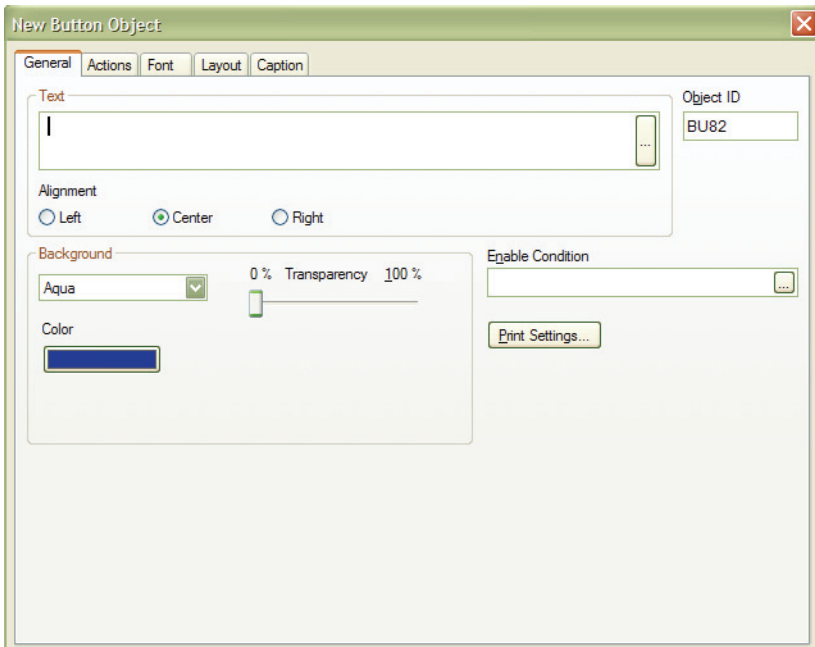


Figure 14. The General tab of the Button Properties

Buttons may look quite different. In the **Background** group, you have the choice between a **Plain** background, **Aqua**, which gives the button a glassy look, and **Image**. If you decide to use an **Image** option you can either use a single image or a combined image consisting of three pictures (for the active, the pressed and the inactive button) in a horizontal row. QlikView supports the image formats bmp, jpg, jpeg and png.

Note: A sample 3-part image file called **QvEYE3.png** for use in the end of chapter exercise is included in the Files folder of materials that you installed with the course materials.



Figure 15. Three different looks for buttons: Plain, Aqua and Image

A newly created button is placed at the top left of the sheet, if that position is free. Buttons can be resized and moved just like any other object.

Actions

Actions can be used in many ways in a QlikView application to trigger different events. For instance Actions can be placed in buttons for users that find it difficult to navigate in the QlikView menus and Toolbars.

Actions can be connected to a number of sheet objects and triggers by pressing the sheet object with the mouse. Actions can be used in the following sheet objects:

- buttons
- text objects
- gauges
- line/arrow objects

In the chapter exercises we will use triggers in buttons and line/arrow objects.

To make use of an Action, we need to create a sheet object that can hold one or several actions. Let us start with a button which is an easy and natural object for an action. We are going to create a button that selects the current year and locks that selection.

In order to do this, we need to perform an advanced search in the Year list box, or more correct in one of the actions that we will connect to the button. In this case we are going to connect an action to a variable holding the value of the current year. Since we want to lock the selection, we need to combine two actions in this button.

- 1 Right click on the sheet and choose Select Fields in the object menu.
- 2 Add the fields Country and Year as list boxes.
- 3 Create a new button and type "Select current year" in the text area.
- 4 Go to the Actions page of the properties dialog

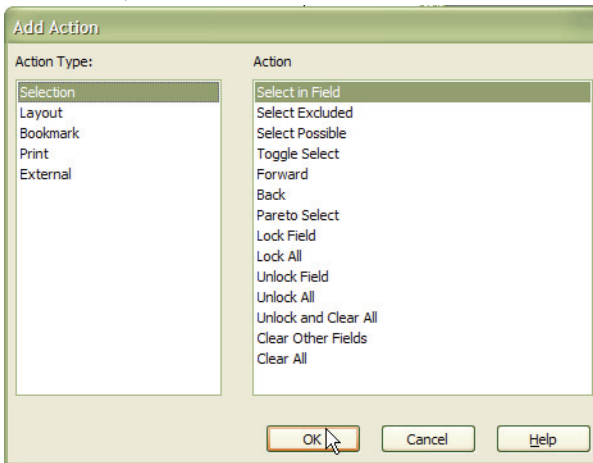


Figure 16. The Add Action dialog

In the Add Action dialog, a number of different Action types may be selected. To each Action type a number of Actions are related.

- 5 Click Add to add a new action.
- 6 Choose the "Select in Field" action.
- 7 In the Field Text box, select the field Year.
- 8 In the Search text box, click on the three dots
- 9 Go to the Variables tab in the bottom part of the page.

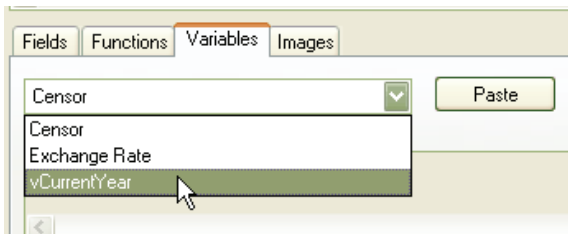


Figure 17. Select the Variable vCurrentYear

- 10 Select the variable vCurrentYear and click Paste.
- 11 Alter the text in the text area so it reads =\$ (vCurrentYear).

- 12 Click OK.
- 13 Click ADD to add a second Action.
- 14 This time, select the ActionType Selection and the Action Lock.
- 15 Click OK to close the Button Properties dialog.

For detailed descriptions of all the different **Action Types** and **Actions**, please refer to either the online help file or the reference manual.

The Line/Arrow Object

The Line/Arrow object is a simple sheet object used to create borderlines and pointers in the layout. You may experiment with line/arrow objects on your own if you like. Remember, too, that Actions can be assigned to text objects, line/arrow objects and gauge charts.

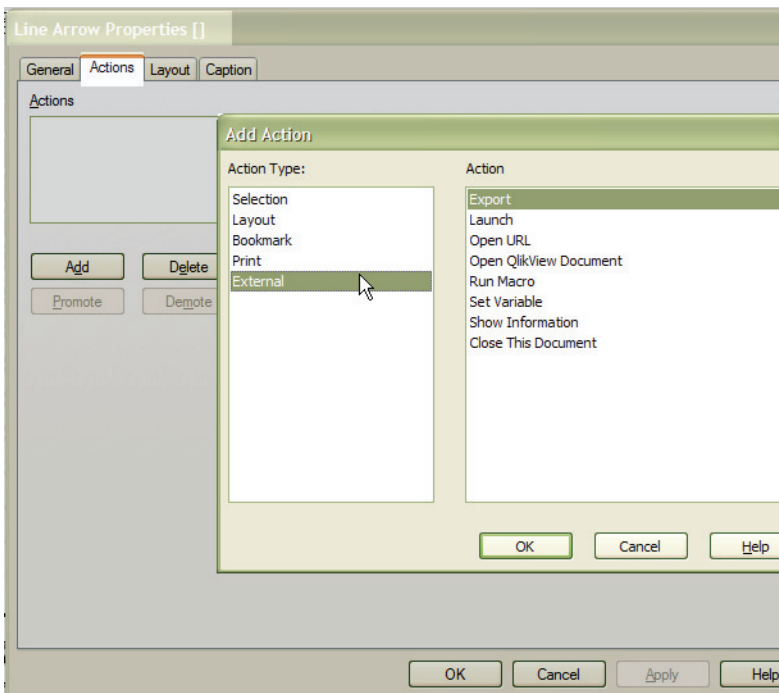
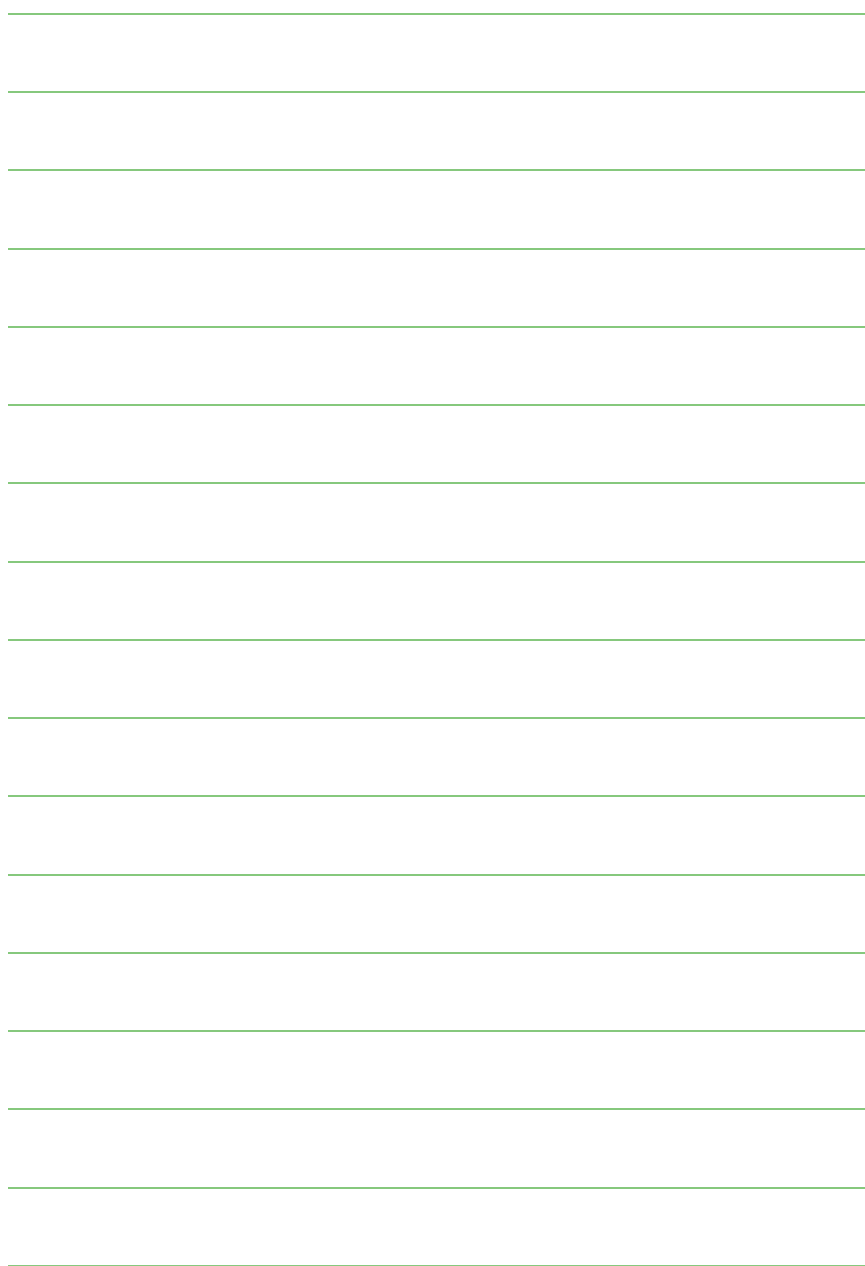


Figure 18. Assigning an Action to a Line Arrow Object





Exercises

In these exercises, you will create a button using a 3-part image and use it to launch the QlikView Help documentation. You will set it off from the surrounding objects using a Line/Arrow object. In the exercises that follow, you will build other buttons and use a Line/Arrow object to initiate an Action.

Do: Help Button

- 1 Navigate to the c:\QlikViewTraining\DesignerII\Chapter6 directory and open the **QVDesignerII_Chapter6.qvw** file.
- 2 Save a copy of the file to preserve the original in case you want to start again from the beginning later. Do this by using the **File | Save As** command. There is also a QlikView file ending in “**_Solution.qvw**” containing the completed exercise for your reference.
- 3 Go to the **Workspace** page and create a new button object. In the text window, type **HELP!**
- 4 For the Background, select **Image** from the dropdown list. Click on the **Select Image** button and navigate to the Files location for the course materials. Click on the **QvEYE3.png** file
- 5 Select the **External** from the **Action Type** list and **Launch** from the Action column. the **Launch** page. In the **Application** dropdown, navigate using the **Browse** button to the: **C:\Program Files\QlikView\English.chm** file. Click **OK**
- 6 Experiment with the look and feel, font choices and placement of the button. When you click it, the QlikView help file should open.



An example of how your HELP! Button might look.

- 7 Experiment with methods for demarcating the button using the Line/Arrow object.

Do: Clear and Unlock Button

- 1 Continue working in the exercise file for this chapter
- 2 Create a button by right clicking on the sheet and select **New Sheet Object**. Select **Button** from the sheet objects menu.
- 3 Type the text “Unlock and Clear” all on the **General** page of the Button properties.
- 4 Go to the **Actions** page by clicking on the Actions tab.
- 5 Click **Add** to add an Action to the Button.



- 6 Make sure that the **Action type** *Selection* is highlighted and select the Action "Unlock and Clear All".
- 7 Press **OK** to activate the Action.
- 8 Click **OK** in the Button object properties to create the button.
- 9 Make a selection in the *Year* list box and lock your selection by right clicking the list box and select **Lock**
- 10 Do the same in the *Country* list box
- 11 Use the button you just created to unlock and clear your selection.

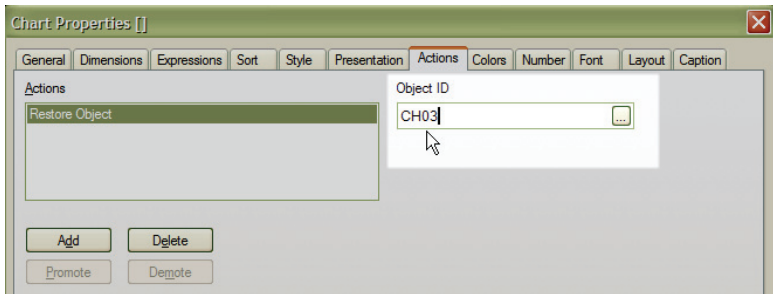
Note: Actions can be used in other sheet objects than buttons. For instance you can use actions on Line/Arrow objects and we are going to create an arrow that will take us back to the previous sheet.

Do: Line/Arrow Action

- 1 Continue working in the exercise file for this chapter
- 2 Create a **Line/Arrow** object by right clicking on the sheet and select new sheet object. Select **Line/Arrow Object** from the sheet objects menu.
- 3 Select **Horizontal** as orientation and a filled arrow going to the left as Arrow style.
- 4 Go to the **Actions** page of the **Line/Arrow** Objects dialog.
- 5 Select **Layout** as Action style and **Activate Previous sheet** as Action.
- 6 Click **OK** and then **OK** again to close the Line/Arrow Object.
- 7 Adjust the size of the arrow so that you only see the arrow head.
- 8 Click on the Arrow head to get back to the previous sheet.
- 9 Go back to the Actions sheet and create an arrow that would take you to the next sheet as well and place the two arrows next to each other on the sheet. Place these arrows on all the sheets in your QlikView exercise file.
- 10 Save your work.

Do: Associating an Action with a Gauge

- 1 Continue working in the exercise file for this chapter
- 2 Right click on the *Margin %* gauge on the Workspace sheet and go to the **Actions** page of the gauge.
- 3 Click on the **Add** button to add an action to the gauge.
- 4 Choose **Layout** from the **Action Type** column and **Restore Object** from the Action column.
- 5 Type "CH03" in the Object ID box (you can locate the Object ID of any sheet object by viewing **Properties | General** or under **Document Properties | Sheets**, among others).



- 6 Click OK. Save your work. Now when you click on the *Margin %* gauge, the *Sales* table opens below it.
- 7 For extra credit, add an action to the gauge to minimize the *Sales* chart when you click on it a second time.

7 OTHER SHEET OBJECTS

Objectives

- Explore other useful sheet objects
- Understand their use
- Build selected objects in the chapter exercises

The Slider/Calendar Object

The slider/calendar object provides an alternative way to select field values. As the name suggests, the slider/calendar object has two completely different modes. As different as they may look, they still work very much the same way behind the visual user interface.

There are examples of a slider and a calendar on the *Other Objects* sheet. They are reproduced, below.

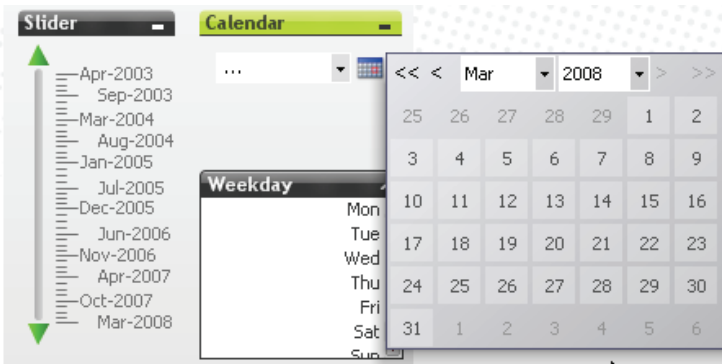


Figure 19. The Slider and Calendar objects

Slider Mode

By dragging a thumb tack along the slider, you can select either a single value, discrete values or a range of values in a field.

A slider for the logical month/year has been included on the sheet *Other Objects* (see Figure 19 The Slider and Calendar objects, above). It may be used for selecting a month or a range of months. Click the slider to get a thumb tack and drag its borders with the mouse to select a range of values. This will determine the width of the thumbtack that can now be moved along the slider.

Calendar Mode

The calendar is used to make selections in fields containing dates. It appears as a dropdown box with a calendar icon to the right. To select values, you can either use the dropdown which works the same way as in a multi box, or you can click on the calendar icon to expand the calendar control. There, it is possible to navigate between months and years by means of the arrow buttons or the **Month** and **Year** dropdown controls. Simply click the dates you want to select or click and drag over a range of dates with the mouse. (See Figure 19 The Slider and Calendar objects above.)

The Input Box

The input box is used for entering values into variables. This is common in budgeting and forecasting applications, for example.

The sheet *Other Objects* in the sample QVW file holds an input box for the variable *Exchange Rate \$ -> €*. Click the current numeric value with the mouse to set the input box to edit mode. Now you can type a new value for the exchange rate and press ENTER to finish the operation. As you can see, the figures for *Sales in €* in the straight table below the input box have changed because the expression for calculating this column is based on the variable value.

This functionality is very useful for keeping calculation parameters up-to-date. It can also be used for powerful what-if-scenarios.

The *Exchange Rate* example is reproduced, below.

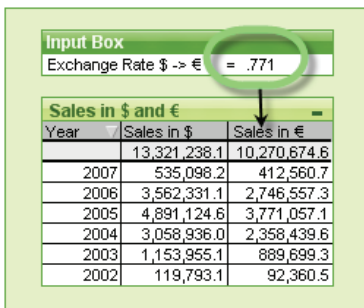


Figure 20. Using an input box to enter a variable used to calculate a value in a chart

The Bookmark Object

Bookmarks are normally managed through the **Bookmarks** menu. To make things easier for the users, you may prefer to display a bookmark object on the sheet(s), just as you might choose to place navigation buttons there. A

sample bookmark object can be seen on the sheet *Other Objects* and is reproduced, below.

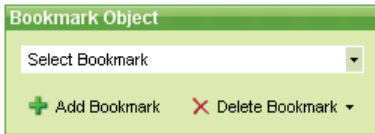


Figure 21. The bookmark object (can be placed directly onto a sheet)

Note: a subsequent chapter covers reports and bookmarks in much greater detail, along with hands-on exercises in the Business Case section of the course.

The Custom Object

This object can contain a custom defined OCX replacement control. Such replacement controls are not created within QlikView, but can be programmed in Visual Basic and imported into QlikView. Given the fact that custom objects are entirely user-defined, they may be applied in a variety of contexts with different purposes and functionality. This functionality is not covered in this course.

The Other Chart Types

So far we have worked with a limited selection of the available chart types; bar, line, pie and block charts as well as the straight and pivot table. Some of the following chart types are perhaps less common, but still very useful in the right circumstances.

Examples of these other chart types can be found in the *Other Objects* sheet of the sample QVW file.

The Combo Chart

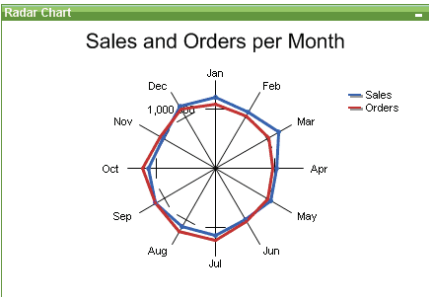
The combo chart is – as its name implies – a combination of two chart types, namely the bar chart and the line chart. This makes it possible to plot sev-

eral expressions in the same chart by displaying some expressions as lines and/or symbols and others as bars.



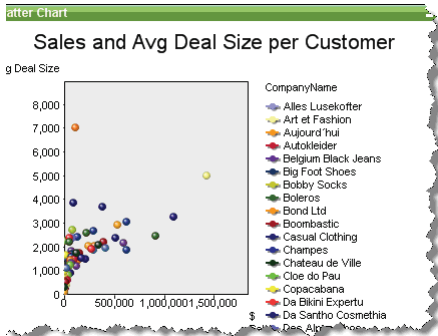
The Radar Chart

The radar chart is equivalent to a line chart where the x-axis is folded around the origin. With several plotted expressions, the resulting chart may resemble a spider's web or radar screen. Radar charts can be used for cyclical, time-based metrics, such as a 24-hour period or a twelve-month period.



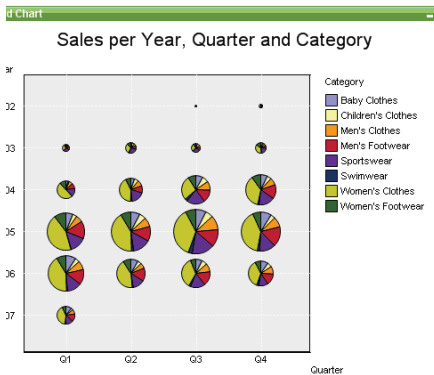
The Scatter Chart

The scatter chart is very useful for plotting the combination of several expressions. This chart can be created using only expressions.



The Grid Chart

The grid chart uses one dimension on each axis, (with an optional third dimension displayed in the plot symbols). The plot symbol itself is determined by the expression.



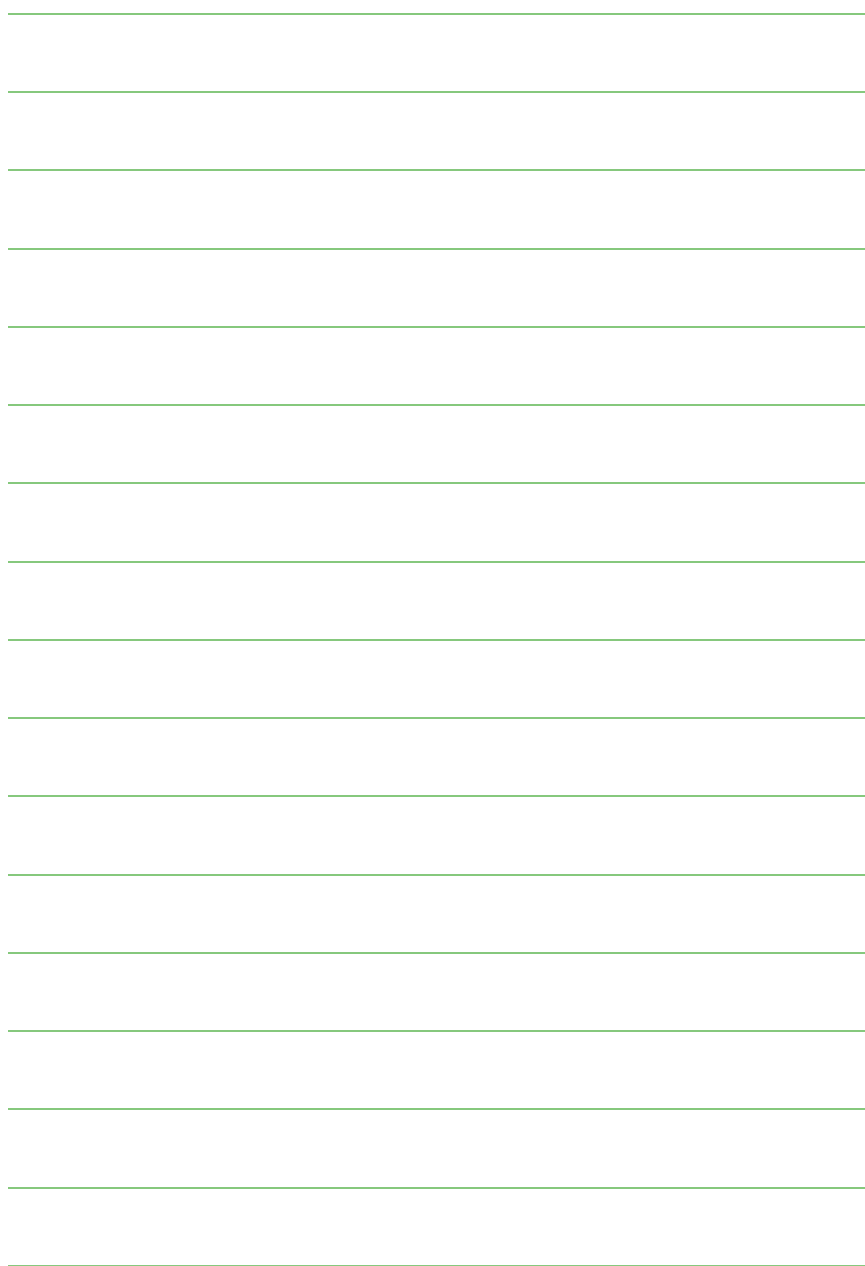


EXERCISE

Do:

To prepare for the conclusion of the course, you will coalesce the knowledge you have gained so far by applying your new skills to implement an actual set of QlikView pages. Now we will take a moment to review some of the key objects from this chapter and create a few on our own, in preparation for that additional work later in the course.

- 1 Navigate to the c:\QlikViewTraining\DesignerII\Chapter7 directory and open the **QVDesignerII_Chapter7.qvw** file.
- 2 Save a copy of the file to preserve the original in case you want to start again from the beginning later. Do this by using the **File | Save As** command.
- 3 As a minimum, be sure to create the following objects for placement on every analysis page:
 - Current Selections Box
 - Statistics Box
 - Consistently placed navigational list boxes (especially relating to the time dimension)
- 4 In addition, create (or copy and then convert) one of your earlier charts into one of the new types we learned about in this chapter:
 - Combo Chart
 - Radar Chart
 - Scatter Chart
 - Grid Chart
- 5 As we learned in one of the previous charting exercises, above, the **Fast Type Change** icon is a powerful way to display the same data in more than one chart type with a simple click of the mouse. Experiment with Fast Type Change on the chart you just created. Do not forget the possibility of using pivot and straight tables (they are charts, too, after all) in your **Fast Type Change** list.



8 ADDITIONAL EDITING IN THE LAYOUT

Objectives

- Learn about advanced properties settings
 - Dynamic chart backgrounds
 - Fast change
 - Title, legend and caption settings
 - Reference lines/text in chart
- Explore the possibilities with Layout Themes

As you get comfortable with the **Properties** dialogs of the various QlikView objects, you will find that they include many possibilities for further improving the layout. To begin with, there are options for settings to the font and color of individual object components, typically charts. There are also preset styles that can be applied. You may even transfer layout settings from one QlikView application to another, which will often shorten the development time significantly.

Dynamic Chart Backgrounds

The background of a chart can now be dynamically calculated. This opens up possibilities for map backgrounds that update with selections, for example. Dynamic chart backgrounds are based on a calculated expression to show dynamic background images that change with the selection. Available for bar, line, combo, scatter and grid charts.

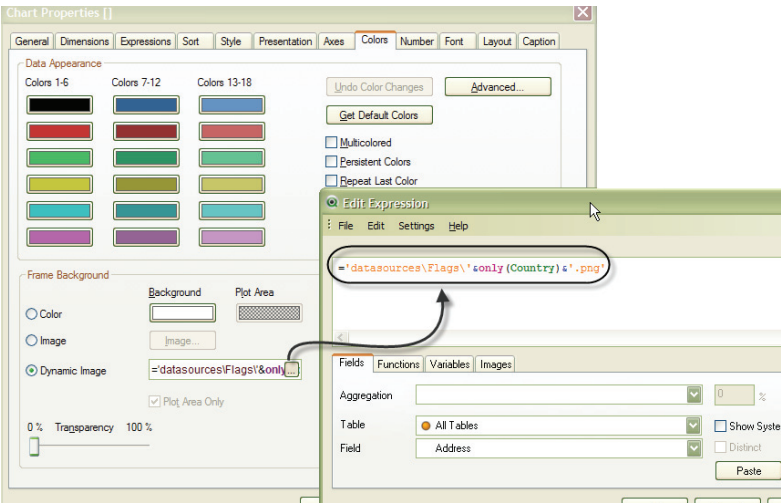


Figure 22. Dynamic chart background image expression for the display of the flag based on the country selected

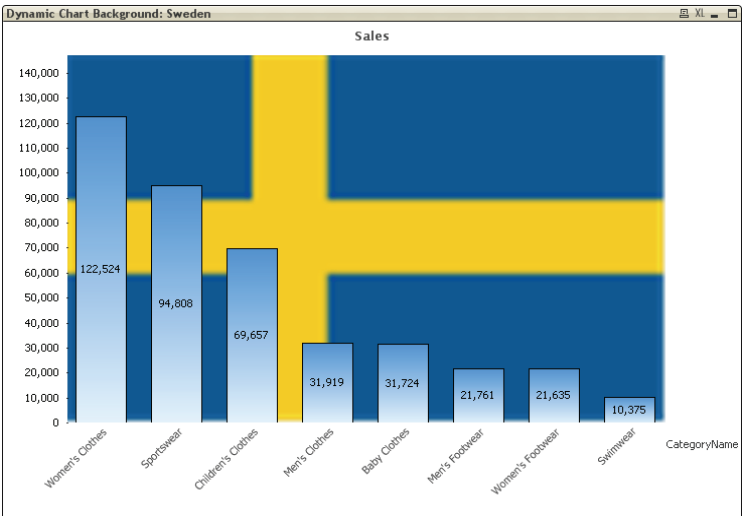


Figure 23. A simple bar chart with the flag displayed dynamically as the country is selected

Title Settings

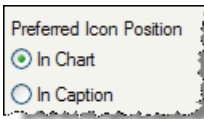
To illustrate, look at the **Chart Properties: General** page for the chart in the example, above in the `QVDesignerII_Chapter8_Examples.qvw` file.

You will notice a button labeled **Title Settings** immediately to the right of the chart title edit box. If we click this button, we are presented with the **Title Settings** dialog. According to the current dialog settings, we are using a standard (classic) title style and a center-aligned title text. The settings for the font are accessed from the **Font** button and the text color can be edited by clicking the corresponding colored button.

Fast Change Chart Type

The **Fast Type Change** option is found on the **Chart Properties: General** page.

You simply check the chart types you would like to admit for fast type change. The icon can be positioned in the caption or in the chart itself. Obviously, when building a “captionless” chart, you would select **In Chart** as the *Preferred Icon Position* (see below).



Tip: When using the fast type change, it may be necessary to make some further layout adjustments for each of the chart types you select. It is a good idea to scroll through them and adjust the layout accordingly for each.

Legend Settings

Navigate to the *Basic Objects* tab in the `QVDesignerII_Chapter8_Examples.qvw` and open the pie chart. This type of chart contains a legend clarifying the key to the colored pie slices. The layout of this legend can be edited through the **Legend Settings** dialog that is

opened from the **Settings** button in the **Chart Properties: Presentation** dialog.

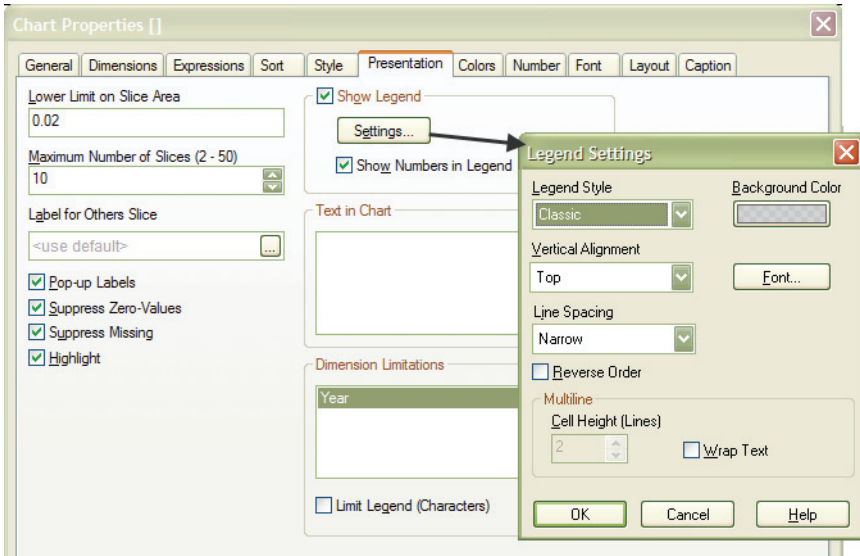


Figure 24. The Legend Settings Menu

Again we may choose between different legend styles, fonts and font colors. There are also options for vertical alignment, line spacing and multi-line text.

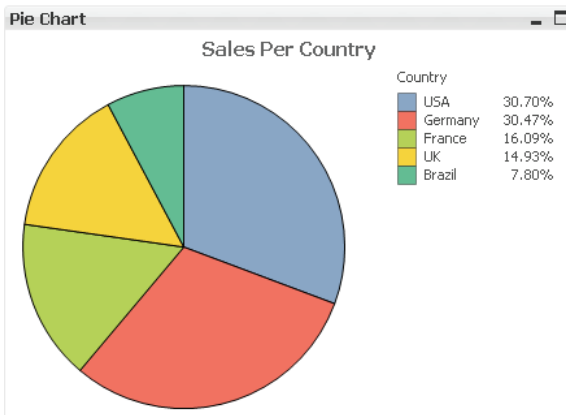


Figure 25. Pie Chart with legend prominent

Reference Lines / Text in Chart

In the **Chart Properties: Presentation** dialog there are two buttons that make it possible to add extra information to a chart.

The **Add... (Reference Lines)** button lets you compare the displayed data to one or more calculated reference lines. There are several formatting options for these reference lines in the **Reference Lines** dialog. Reference lines are not available in all chart types.

The **Add... (Text in Chart)** button lets us add text anywhere we like in the chart. This feature can be used to enter additional information or remarks of any kind. Chart texts can be repositioned by clicking and dragging with the mouse when the chart is in edit mode (CTRL + SHIFT are held).

Tip: as we saw in the Gauges exercise, adding an expression-based Text in Chart label is a powerful means of communicating the result of a calculation both in the chart itself, and in the label. Informative text labels are useful, too.

Caption Settings

Any QlikView sheet object may have a caption and most QlikView objects are created with one by default. When activating the **Properties: Caption** page for a given object, you will find that the option **Show Caption** can be switched on or off.

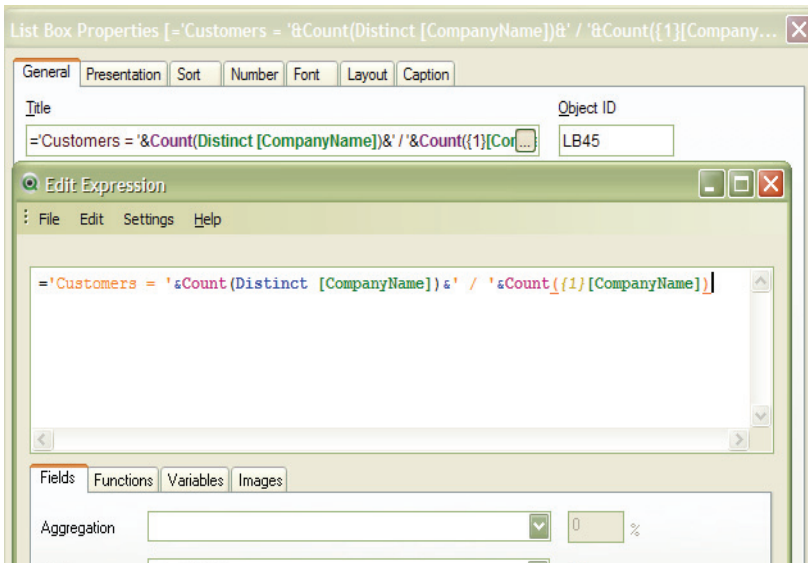
In the **Caption Settings** page you can define non-standard colors for the background and text for the active and inactive state, respectively. You can also specify a font and set the text alignment. There are also options for vertical alignment, line spacing and multi-line text. This dialog holds the formatting options for scrollbars as well. Finally, you can display several caption icons that govern such actions as **Print**, **Send to Excel** and **Clear**, among others. Most of these topics have been covered earlier in the course.

Captions can also contain expressions that can reflect the selection status. A common example involves counting the number of items selected compared to the total value.

Navigate to the *Workspace* tab and open the *Customers* list box. If you make a selection in it, you will notice that the caption displays the number of customers you selected.



Figure 26. This is accomplished with a formula, as below



Using Layout Themes

A theme is a special file containing layout and formatting settings that can be applied on various levels: to a whole QlikView document or to sheets or to individual sheet objects. Whatever you need to do. Themes can significantly reduce time and effort needed to make a QlikView document look nice and consistent. A theme file is in standard xml format and can be viewed in a text editor such as Notepad.

When QlikView is installed, several .qvt files are installed in the Themes folder. These pre-built theme files should be regarded as examples. The main strength of this feature is the ability to create and apply your own themes. You can define themes to comply with different company profiles or visual guidelines and then rapidly deploy them across your development environment.

Applying a Theme

You may apply an existing theme to one sheet object at a time, by clicking the **Apply Theme** button from the **Properties: Layout** page of a single sheet object. Browse for the desired qvt file in the dialog and open it to apply the theme.

To apply a theme to a sheet or to the whole QlikView document, you start from the **Sheet Properties: Layout** or the **Document Properties: Layout** page. Note that a theme may not apply to a certain object type unless it was specified at the time of its creation.

Creating a Theme

While it is possible to create a theme for a single type of sheet object, you will generally want a theme to contain formatting for several or all kinds of objects. The **Theme Maker Wizard** assists you in the creation of themes.

To create a theme you should first of all format the document to your liking and create nice-looking prototypes of all sheet objects you want to include in the theme.

After this, start the **Theme Maker Wizard** from the **Tools** menu. In the first step you have to define a file name and location for your theme. Then you choose the first of your prototype objects as source (you may also choose the whole document or a sheet to include its properties in the theme file) and you select all properties to be included in the theme. The properties are portioned up into different sections: **Object Type Specific**, **Caption & Border** and **Print Settings**. Finally, you define which object types the properties should be used for when the theme is applied.

If you want properties from more object types to be included, you will have to run the **Theme Maker Wizard** repeatedly to add these to your theme.

Tip: You can also change caption and border properties quickly without using themes. Try the **Apply To:** button found on the **Layout** tab of sheets and sheet objects.

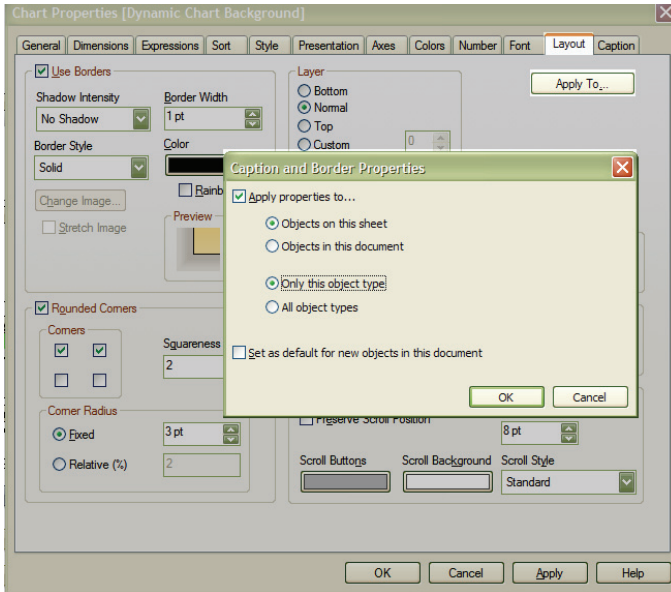


Figure 27. Apply Caption and Border Properties Dialog



Exercises

Although this chapter was primarily theoretical, designed to give you ideas about additional capabilities in the layout of your QlikView documents, and even though other parts reinforced topics introduced earlier like **Fast Change** chart type and **Text in Chart**, it is nonetheless important to pause for a brief review exercise before continuing on to the next chapter.

Do:8

- 1 Navigate to the c:\QlikViewTraining\DesignerII\Chapter8 directory and open the **QVDesignerII_Chapter8_Themes.qvw** file.
- 2 Save a copy of the file to preserve the original in case you want to start again from the beginning later. Do this by using the **File | Save As** command.
- 3 Pick a sheet to work on, navigate to it, and right-click.
- 4 Select **Properties** and go to the **General** page.
- 5 Click on the **Apply Themes** button and select a new theme to apply to your QlikView file. You can also apply themes to individual objects. If you are happy with the theme you applied, go ahead and convert your entire QlikView file to it. If not, close the file and open the backup you created at the beginning of the exercise.
- 6 Open the **QVDesignerII_Chapter8_Examples.qvw** to explore the objects presented in this chapter, including Dynamic Chart Backgrounds and Title, Legend and Caption settings.

Tip: If you are having problems locating the.qvt theme files on your computer, click on the **Settings** menu, choose **User Preferences**, and navigate to the **Locations** page for a display of the different *Resource* Locations. You will see a listing there for *Themes*.

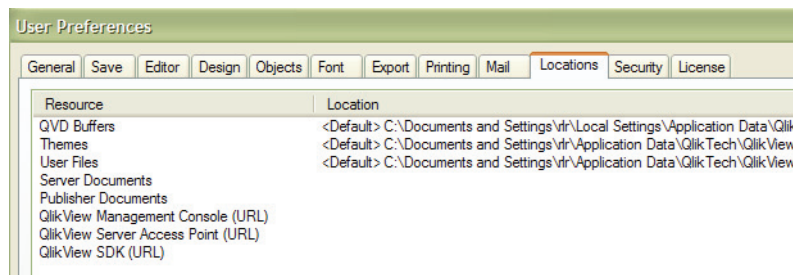


Figure 28. User Preferences | Locations




9 OTHER USES OF EXPRESSIONS

Objectives

- Demonstrate the power of interface design using expressions and dynamic labels
- Define color expressions
- Use some of these types of expressions in the chapter exercises

Expressions are in no way limited to chart expressions and sort expressions. Other uses of expressions are as label expressions, conditional functions, dynamic text in text objects, chart attribute functions or color functions. We will discuss some examples in this chapter.

In the QlikView dialogs, it is easy to determine where expressions may be used, because the edit box will display a special button with the ellipsis symbol  that leads to the **Edit Expression** dialog.

In terms of syntax, you should keep in mind that, in contrast to chart main expressions, calculated formulas must be preceded by an equal sign (=).

Calculated Labels and Dynamic Text in Text Objects

Where you can enter a label text, you can usually also enter a label expression. As stated previously, there are a few objects with calculated labels displayed in the sample QVW file on the *Other Objects* and the *Charts* sheet. On the *Charts* sheet, the title of the list box *Customer* has been given a label expression. When a single country is selected, the caption text will change. Also notice the text objects that appear when a single customer is selected (this is explained in the Conditional Functions section, below).



Selection: 1 customer(s) out of 92

The customer Art et Fashion
in France has purchased
products for 6,2705.08

Figure 29. Charts tab objects with calculated formulas

Using expressions, they get the additional functionality of displaying dynamic text, i.e. text that changes according to the current logical state of the application.

Conditional Functions

Conditional functions are useful to us in different situations:

- They can determine if a sheet object should be hidden or displayed (**Show - Conditional** set in the **Properties: Layout** page).
- They can determine if a chart or table should be calculated or not (**Calculation Condition** set in the **Chart Properties: General** page).
- They can determine if a button should be enabled or not (**Enable Condition** set in the **Button Properties: General** page).

When you select a single customer in the *Charts* sheet, a previously hidden text object will show text information about this customer. The display of this text object depends on whether there is a distinct customer selected or not. By examination of the properties of the text object, you should be able to figure out how similar results can be achieved in your own applications.

The conditional display condition for the text object just described is shown in the figure below (Sheets can also be made to show conditionally, but the conditional expression is entered on the **General** tab then):

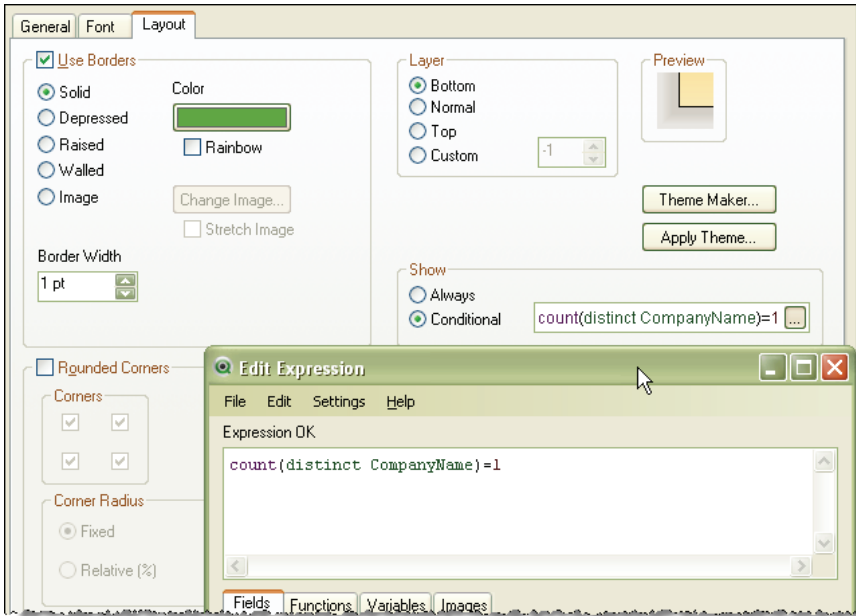


Figure 30. Setting the conditional display expression for a text object

Chart Attribute Expressions

The chart attribute expressions are accessed by clicking the expansion icon in front of a chart expression in the **Chart Properties: Expressions** dialog. Attribute expressions may be used for a dynamic formatting of the expres-

sion data. Their use depends on the chart type and the display options. You can read more in the QlikView Reference Manual or the Help file.

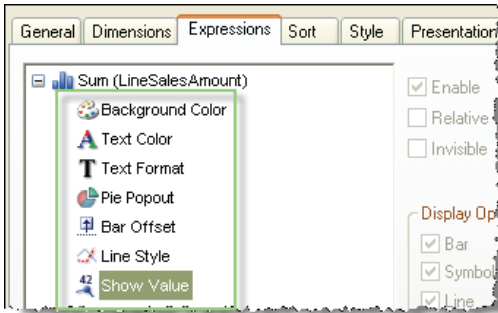


Figure 31. Chart Attribute Expressions

Color Expressions

In nearly all instances where you can define a color for an area or background in a sheet object you do that in the **Color Area** dialog. There you may define a solid color, a color gradient or a dynamic color, calculated by means of a color function. Please refer to the QlikView Reference Manual or the Help for details.

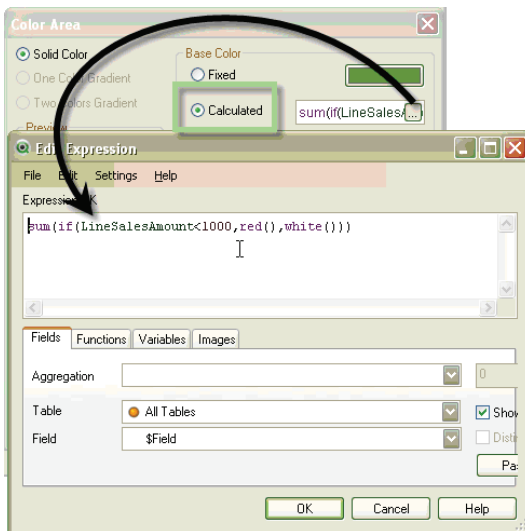


Figure 32. Calculated color expression

Calculated Fields in List Boxes

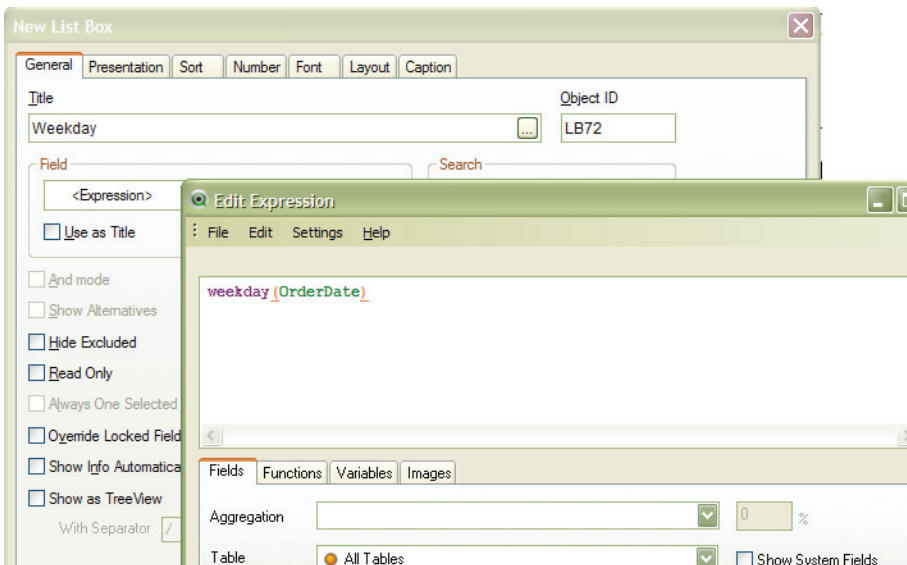
It happens that you would like to have a list box showing a field which has not been created by the QlikView script when loading the data. In this case, you have the possibility to use an expression to calculate the values based on existing fields. A typical example is a field combining the values of two existing fields. Calculated fields are defined by selecting *<Expression>* from the dropdown list **Field** in the **List Box Properties: General** page.

For example, you might wish to determine the weekday name for a given date field.

By creating a new list box based on the expression:

=weekday (OrderDate)

Where *OrderDate* is the date you have chosen —



You could create a field listing the days of the week, even if those data were not part of the original source, e.g. loaded by the QlikView script, and dis-

play it in a fully functional list box, like the one below (located on the *Other Objects* sheet).



Calculated Dimensions in Charts

Likewise for charts, calculated dimensions work the same way as calculated fields in list boxes. They create a field that does not exist in the document and then use it as a dimension in a chart. To define a calculated dimension, click **Add Calculated Dimension** in the **Chart Properties: Dimensions** page. The process is identical to the one we just used to create the *Weekday* list box.

The Expression Overview Dialog

The **Expression Overview** dialog lists all expressions within a document allowing for easy modification. The main benefit is that the user may edit numerous expressions at once. To open this dialog, select **Expression Overview** from the **Settings** menu.

By using the check boxes in the top left corner, you can modify the content of the expression list. Just deselect the expression types you do not want to have included in the list.

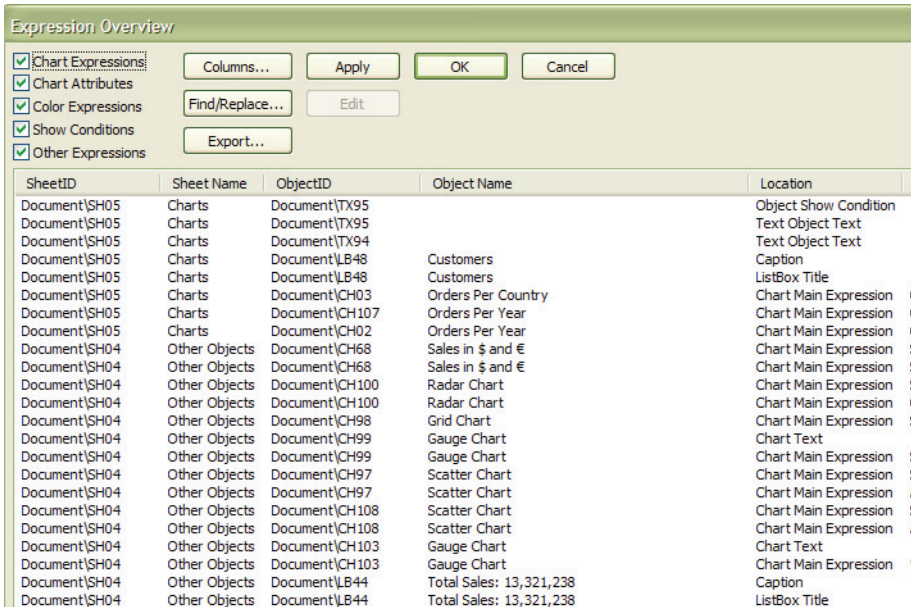


Figure 33. The Expression Overview dialog

The **Columns** button can be used to add or remove columns displayed in the overview.

The **Find/Replace** button allows the user to search for and replace an expression.

Tip: remember to click on **Apply** to confirm any changes or edits you have made using the **Expression Overview** interface!



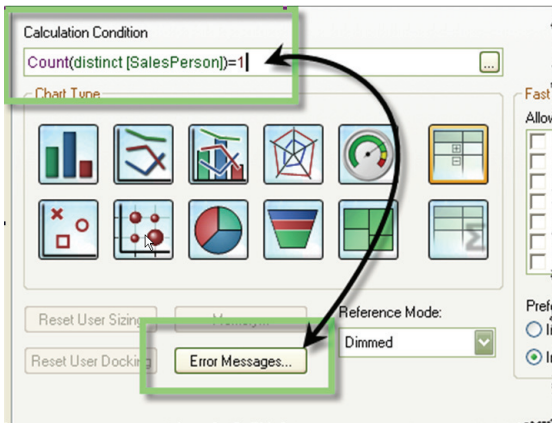
Exercises

The following brief exercise focuses on one central example for calculation conditions, useful when you want to force a user to make a selection before a sheet object displays.

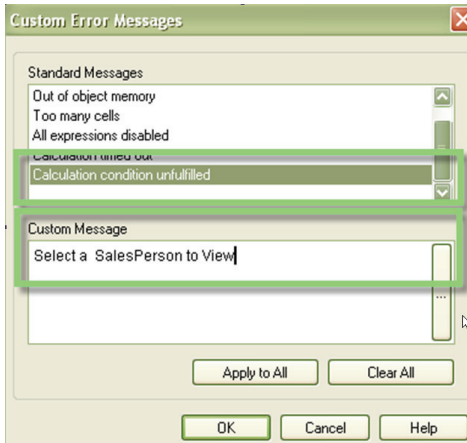
Do:

- 1 Navigate to the c:\QlikViewTraining\DesignerII\Chapter9 directory and open the **QVDesignerII_Chapter9.qvw** file.
- 2 Save a copy of the file to preserve the original in case you want to start again from the beginning later. Do this by using the **File | Save As** command. There is also a QlikView file ending in “_Solution.qvw” containing the completed exercise for your reference.
- 1 Navigate to the *Workspace* tab.
- 2 Copy the pivot table from entitled **Sample Pivot Table** and modify the dimensions so that it contains the following fields:
 - *Country*
 - *CompanyName* (Labeled as *Customer*)
 - *Year*
 - *Quarter*
- 3 Next, move to the **Expressions** tab and **Enable** the *Avg Deal Size* expression (the *Sales* expression should already be enabled).
- 4 On the **Presentation** tab, make sure **Show Partial Sums** is checked for the *Country*, *Customer* and *Year* fields.
- 5 Click **Apply**.
- 6 Return to the **General** tab and type the following into the **Calculation Condition** box in the left-center of the page:


```
Count(distinct [SalesPerson])=1
```



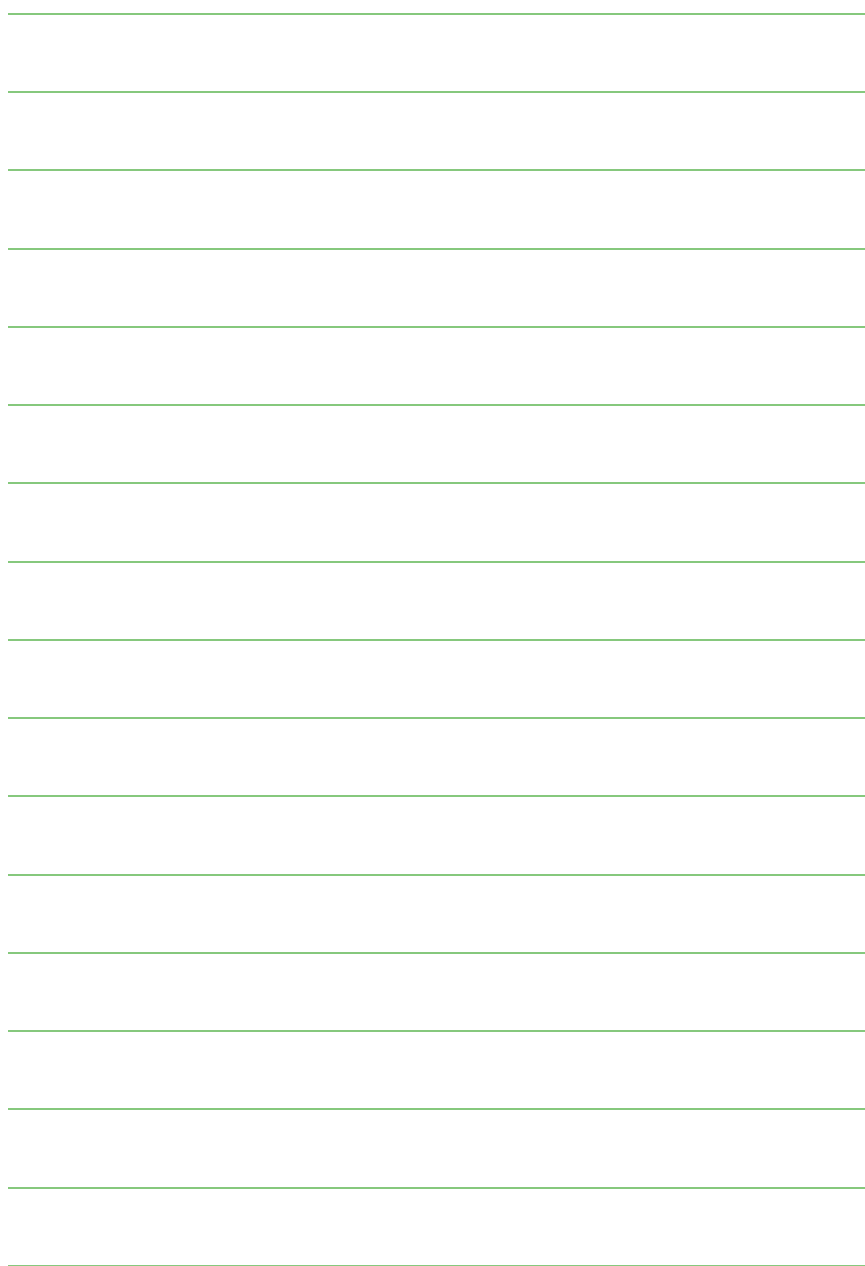
- 7 Click on the **Error Messages...** button and type in a **Custom Error Message** for the status *Calculation condition unfulfilled*, as below:



- 8 Click **OK** and then **OK** again.
- 9 Clear your selections. You should see the following until you select one *SalesPerson*:

10 Save your work.





10 REPORTS AND BOOKMARKS

Objectives

- Introduce QlikView reports
- Review basic reporting functionality in QlikView
- Build a basic report in the chapter exercises

This chapter introduces the main concepts of QlikView reports and bookmarks. After an overview section for each of these tools, the student will have a chance to create and edit both a report and a bookmark in the chapter exercises.

Introduction – Reports

Some people might consider the entire QlikView document a report, since each sheet contains dynamic objects and can be quickly printed or captured to a view.

That said, in QlikView, printing a report often means printing a single table or graph. This is easily accomplished by selecting a sheet object and then choosing Print from a menu or toolbar.

Sometimes however, you need to produce more complex reports including multiple charts and/or tables. This is where the QlikView Report Editor can be a powerful ally.

The QlikView Report Editor offers the possibility to group several different sheet objects together on one or more pages with full control of page layout, including headers/footers etc.

Note: When using the term *Report Generator* one usually refers to a well defined category of software products. These typically work by combining data from several SQL queries (more or less obscured from the user by means of graphical interfaces). Each of these queries, in turn, might be formatted for printing in different ways. QlikView reports fetch their data from QlikView sheet objects, which, in turn, are populated by the QlikView internal database. They are not dependent, therefore, on external SQL queries. Just as it is absolutely correct to say that QlikView can produce reports, it is incorrect to label the QlikView Report Editor a report generator in the traditional sense.

QlikView also lets us define bookmarks that are used for preserving, sharing and restoring specific selection states of the document for future reference. Bookmarks are introduced later in this chapter.

QlikView Report Types

Here are the types of QlikView Reports (see **Note**, below):

Document Reports

Document reports are created with the QlikView document and stored as part of the .qvw file. Any user accessing the QlikView document locally or via QlikView Server can access the document reports in the document.

User Reports

User reports are created by a user working with a document from QlikView Server. The report is stored on the local client machine in a manner similar to that of user bookmarks. Only the local user has access to his/her own user reports. User reports can only be created from the Windows based QlikView clients (not from Java or Zero-footprint clients).

Personal Server Reports

Personal server reports are only available when working with a document on QlikView Server and only to authenticated users. They are stored in a repository on the server and are accessible to the user from any computer where authenticated.

Shared Server Reports

Shared server reports are only available when working with a document on QlikView Server and only to authenticated users. Any user who has created a personal server report may flag this as shared to other users. It will then become available to other users. Just like personal server reports, shared server reports are stored in a repository on the server.

Note: Since this is a course for QlikView Designers, the possibility of connecting to a QlikView Server is not assured. Thus, we will focus on creating and editing document reports in the chapter exercises.

Managing Reports

Reports are displayed in a list in the **Reports** menu. If you want to print one of these reports, you can simply select it there. This will open the **Print** dialog with the usual print settings.

QlikView reports can be printed by an ordinary, installed printer or alternatively by the QlikViewPDF printer, which is normally installed automatically together with QlikView.

To add new reports or to edit existing ones, choose the command **Edit Reports** from the **Reports** menu. You will get to the **Report Editor**, which consists of two pages: The **Report List** and the **Page Editor**. When first opening the **Report Editor** the **Report List** will be shown. It is used for managing reports. You can create new reports, delete existing reports or select a report from the list for further editing. The list shows **Document Reports** or **User Reports**, depending on what type is selected in the dropdown list at the top.

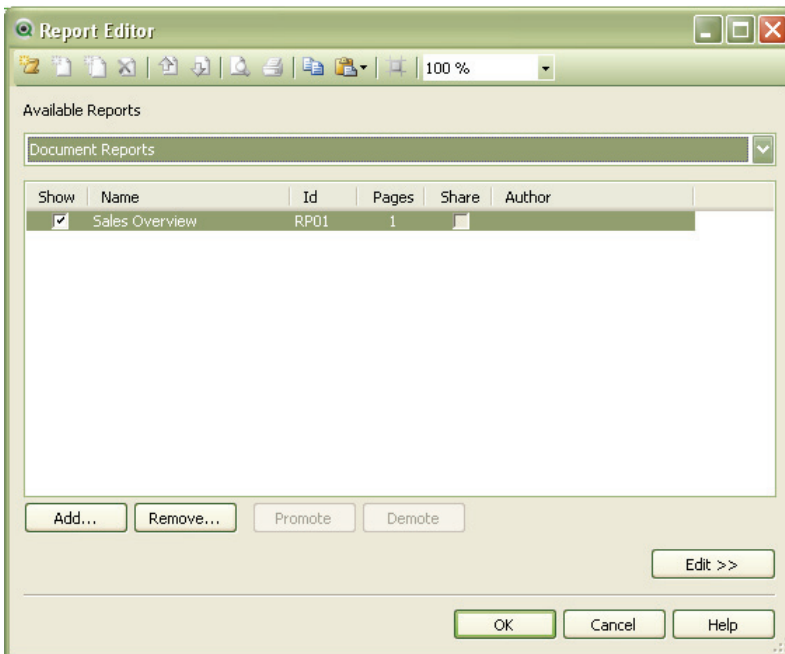


Figure 34. The Report List dialog

Creating and editing reports

The dialog page **Report Editor** is used to define the pages of the report selected in the **Report List**.

The page has two panes, several settings and a toolbar to assist you in designing your report. On the left, a vertical pane (page list) is shown. On the right is a larger pane (preview) where you can edit the report page currently selected in the page list.

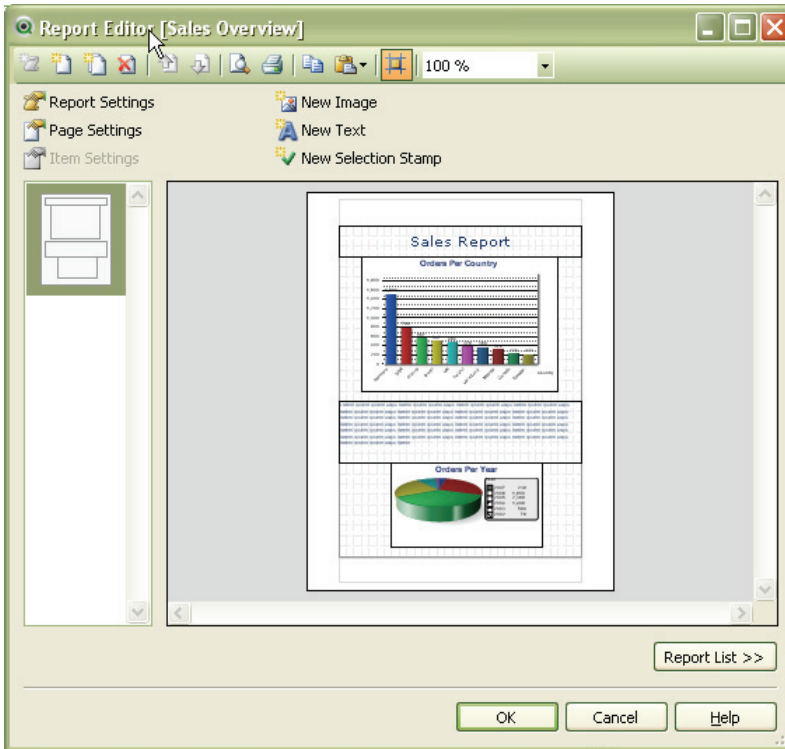


Figure 35. The Page Editor dialog

The toolbar contains a number of buttons for managing the page list. Use them to add pages, to delete selected pages from the page list or to change their order. You can have a print preview of your report and print it out. The **Snap to Grid** function helps you to position the objects more accurately on the page. The **Copy** and **Paste** commands make report editing more comfortable and refer to objects as well as to entire report pages.

Clicking **Report Settings**, **Page Settings** and **Item Settings** will open settings dialogs for the entire report, the selected page or the active item on the current page. These hold some additional settings and adjustments such as the report title or page orientation.

Tip: Important enhancements in the latest version of the **Report Editor** include:

- The design grid
- Alignment tools
- Multi-object drag and drop
- Configurable section spacing
- Show condition on reports
- Calculated report names
- Zoom control
- Tabbed navigation across report objects

Feel free to experiment with these features, along with the more basic reporting capabilities in QlikView's **Report Editor**, as you complete the chapter exercises.

There are two different types of report pages:

Single Paper Pages

can contain any number of sheet objects and objects may overlap each other. This page type is always printed on exactly one paper page (or PDF page). To fit the page, sheet objects are zoomed or truncated if necessary.

Multi Paper Pages

can only display a single sheet object, but this object can be stretched over as many papers (or PDF pages) as are necessary. Multi paper pages are appropriate for printing longer tables.

When a page is selected in the page list, it is displayed in the preview pane to the right. This pane is also employed for editing the page layout. When a new page is added, it is at first completely blank. Adding picture elements; sheet objects, text or images to the page is incredibly simple.

Any sheet object can be copied from the QlikView sheets directly to the pane, by clicking and dragging. The placement and sizing of elements can subsequently be edited in the usual manner.

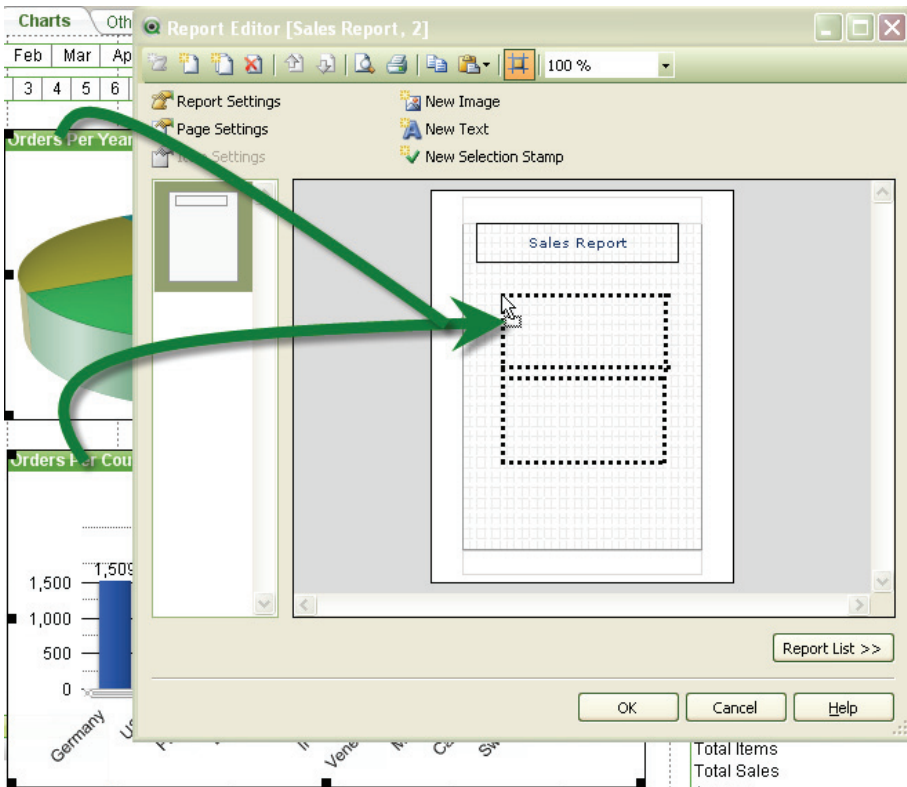


Figure 36. Transferring two charts to the Report Editor simultaneously

As the report is dynamically connected to the QlikView document, the looks of the various objects in the report will always be updated for data contents and current selections in the QlikView document at the time of the printout or viewing.

Reports may be complemented with additional images, texts, or with a selection stamp showing the current selections. To enter such elements click the respective button in the upper part of the **Page Editor** dialog.

Introduction – Bookmarks

Bookmarks are used for preserving certain logical states in QlikView documents so that they may be recalled later on. As with reports, there are different types of bookmarks for different situations.

There are four main types: Document Bookmarks, User Bookmarks, Personal Server Bookmarks and Shared Server Bookmarks.

Bookmark Types

Here are the different types of bookmarks in greater detail. Please also note that, since this is a course for QlikView Designers, the possibility of connecting to a QlikView Server is not assured. Thus, we will focus on creating and editing User Bookmarks in the chapter exercises.

Document Bookmarks

are stored inside the .qvw document. They will always be available to whoever opens the document locally or from a QlikView Server.

User Bookmarks

are stored separately on the user's computer. They will only be available to the user who created them on the computer where they were created. If the document is moved or renamed all personal bookmarks related to it will be lost. In QlikView versions before QlikView 8, this was the only type of bookmarks that a user could create when working with a document on QlikView Server. Although still supported, they are largely made obsolete by the server bookmarks in QlikView 8.

Personal Server Bookmarks

are only available when working with a document on QlikView server and only to authenticated users. They are stored in a repository on the server and are accessible to the user from any computer where authenticated.

Shared Server Bookmarks

are only available when working with a document on QlikView Server and only to authenticated users. Any user who has created a personal server bookmark may flag this as shared to other users. The bookmark will then become available to other user. Just like personal server bookmarks shared server bookmarks are stored in a repository on the server.

Managing Bookmarks

QlikView bookmarks are managed from the **Bookmarks** menu. Document bookmarks are listed in the menu under **Document Bookmarks**.

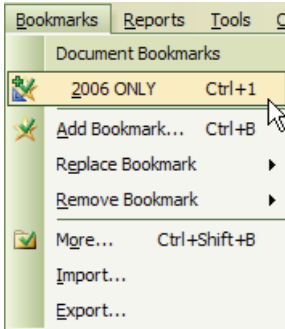


Figure 37. The Bookmarks Menu

Tip: You can replace an existing bookmark with the current selection state by using the **Replace Bookmark** command.

When advanced management is required, the commands **More**, **Import** and **Export** are employed, either from this menu, or by selecting **More** to open the **Bookmarks** dialog

The *Bookmarks* dialog can be opened by clicking **More** in the **Bookmarks** menu. It is divided into two pages (local documents) or four pages (server documents), one for document bookmarks, one for user bookmarks, one for

your personal server bookmarks (server documents only) and one for other users' shared bookmarks (server documents only).

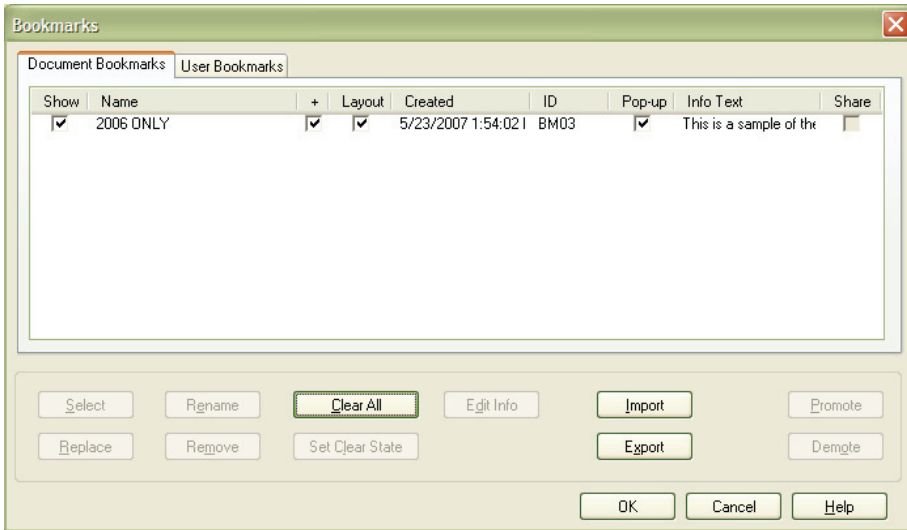


Figure 38. Managing Bookmarks

Please note that bookmarks can be imported and exported from this dialog, as well.

Creating Bookmarks

The **Add Bookmark** dialog includes:

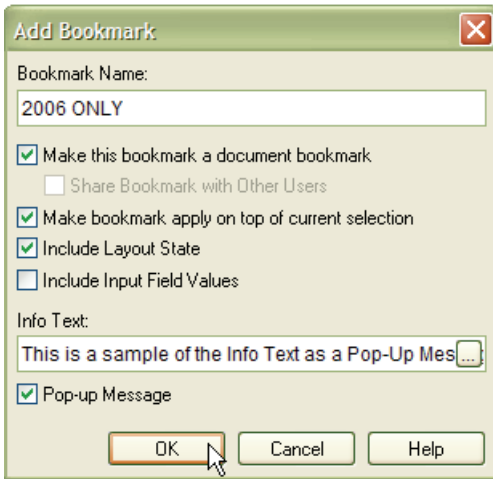


Figure 39. Add Bookmark dialog

Check the alternative **Make this bookmark a document bookmark** to save it with the document, else a user bookmark is generated.

If **Make bookmark apply on top of current selection** is checked, the bookmark will be applied without the current selections being cleared first.

The alternative **Include Layout State** saves the layout state for the sheet objects of the current sheet with the bookmark (also increasing the size of the bookmark file).

Include Input Field Values takes into consideration the possibility that the values entered by a user into an input field be reflected in the logical state preserved by the bookmark.

Mailing Bookmarks via QlikView Server

When you find something worth sharing in a QlikView Server document, you may, with just a single command from the menu, send a snapshot to another user. A temporary server bookmark (including layout status information) will be created on the server. A mail with a qvp URL encoding the bookmark reference will be generated. You just need to enter a recipient name and any text you like to go with the link. When the recipient clicks on the link, the QlikView Server document will be opened and your selections and layout status reapplied. This functionality obviously relies on the recipi-

ent having access to the document on the server and access to relevant parts of data and layout therein.

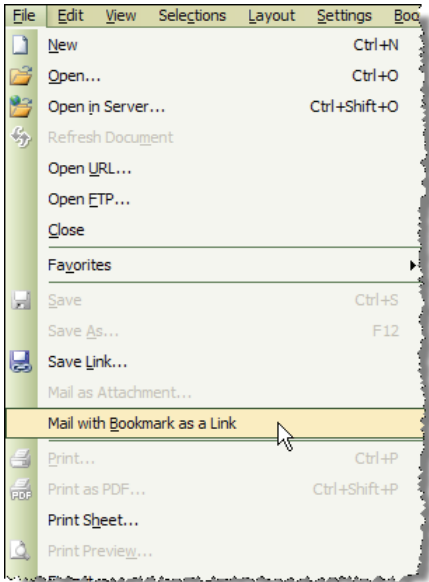
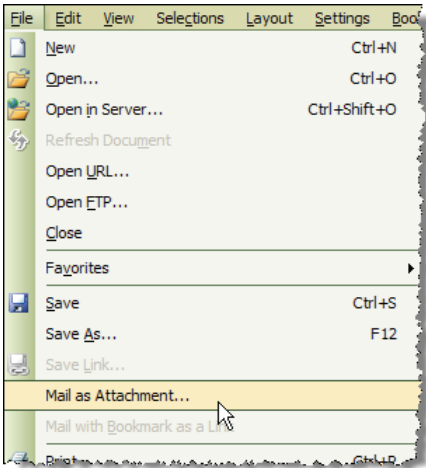


Figure 40. Mailing a Bookmark via QlikView Server

E-Mail as Attachment, Other Options

New in QlikView 8 is the **Mail as Attachment** command on the **File** menu. It is only available when working with a local document (otherwise you would use the server related options discussed earlier). This command creates an e-mail with a copy of the current .qvw document attached. The mail recipient will be able to open the .qvw document provided he has access to QlikView and access rights to the document (if Section Access security is used). Obviously, you have to have a mail client configured for this command to work.



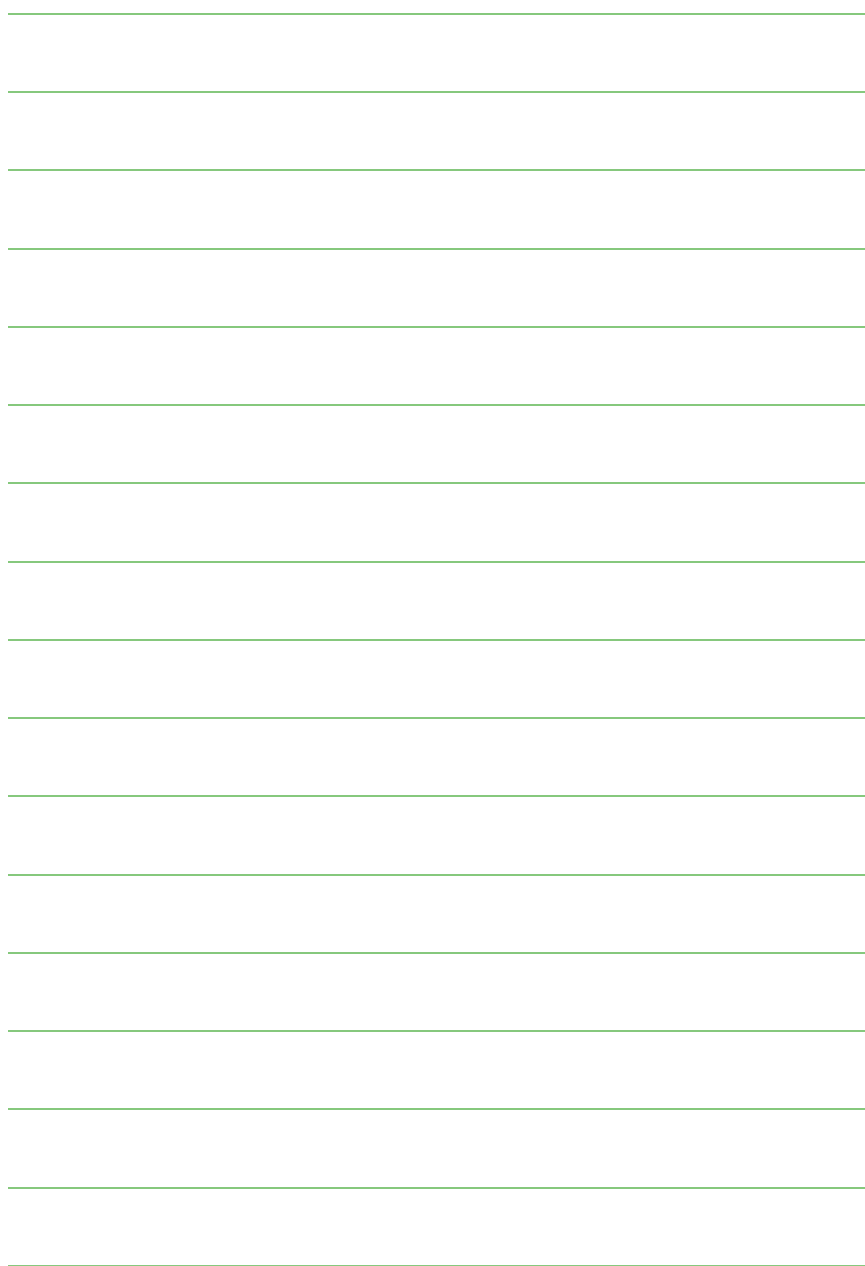
The advantage here for some user environments is that no QlikView Server is required or involved. The entire QlikView file is automatically attached to an e-mail, ready for you to address and send in a single click.



Exercises

Do:

- 1 Use the **Report Editor** to create a QlikView report consisting of a few “papers” incorporating some QlikView objects from the sample document and add some suitable text. Feel free to utilize options in the **Report Settings**, **Page Settings** or **Item Settings** dialogs.
- 2 Make some selections and generate both some document bookmarks and some personal bookmarks. Delete the selections and retrieve the bookmarks via the **Bookmarks** menu. Then save the bookmarks to the desktop as a bookmark file. Now, remove all bookmarks, verify that they are gone and finally import the .qbm file and retrieve the bookmarks again.



11 BUSINESS CASE WORKSHOP

Objectives

- Build a working QlikView analysis application
 - Review business requirements
 - Review data sources
 - Plan development effort
 - Build prototype
- Review and discuss with instructor/students

The QlikView Designer Exercise tests the student's ability to create a QlikView file employing a list of required objects. Students will review their work with the instructor at the end of the class.

Exercise Files

There are two required files for the Exercise. They are data sources to be used in the creation of the application.

The files include:

- Employee.xls
- Sales.xls

Exercise Scenario

Company

The Company in this scenario sells apparel around the world. The QlikView application to be built in the exercise is a basic sales analysis tool designed to present sales information in terms of

- Time
- Country
- Salesperson
- Product
- Customer

The sales data is supplied with the exercise. Loading it into your QlikView application is part of the exercise.

Business Requirements

The Company in this scenario is looking for a business intelligence tool to provide analysis for sales. The business intelligence tool should provide summary to detail level analysis through an easy-to-use, yet robust and flexible

interface capable of handling multiple levels of user skill and requirements. The specific business requirements are listed below.

About Time

The Company is interested in analyzing their data over time. Time should be displayed based upon the *OrderDate* field in the dataset. Analysis should be possible based upon *Year*, *Month* or *Week* values for the entire dataset.

Dashboard

The Company requires a dashboard showing an annual sales comparison, sales by country and seasonal sales (by month), as well as the ability to make selections based on values in the following fields:

- Year
- Month
- CustomerCountry
- Customer
- Product
- Salesperson

The Dashboard should also show what selections have been made (the current selections) at all times.

Sales

A Sales Analysis section should be built. At minimum, it should include the following tools:

- 1 **KPI Margin Goal** table: with detail for Customer, Product, Salesperson and Country with Key Performance Indicators (KPIs) for Sales, Margin Percentage, Number of Orders and Average Order Value.
- 2 **Top Customers** chart: showing KPIs for Sales, Margin and Margin Percent
- 3 **Order Details** table: showing Customer, Order ID, OrderDate, and Product and KPIs for Sales and Quantity
- 4 The Sales Analysis section should also include the ability to make individual selections based on values in the following fields:
 - Year
 - Month
 - CustomerCountry
 - Customer
 - Product
 - OrderID
 - OrderDate
 - Salesperson

The Sales Analysis should also show what selections have been made (the current selections) at all times.

Details

A Detail section should be built. At minimum, it should include the following tools:

- 1 **Top Cycle Group Fast Change Chart:** with detail for Customer, Product, Salesperson and Country and KPIs for Sales, Margin and Margin Percentage
- 2 **Product Details Table:** including detail for Customer, OrderDate, ProductID and Product and KPIs for Sales and Quantity. Subtotal by Customer, OrderDate and ProductID. Add a Fast Change capability to this table to switch between pivot and straight modes.
- 3 **Sales Detail:** a box listing statistical information about the Sales KPI that changes based on the selection status in the application and includes the:
 - Total number of orders
 - Total Sales
 - Average Sale Amount
 - Minimum Sale Amount
 - Maximum Sale Amount
- 4 **Margin Gauge:** A gauge that shows the Margin Percent KPI based on the selection status. The gauge should include demarcations for the target Margin Percentage and clearly delineate which values fall above or below that point. Ideally, the gauge will also display the Margin Percentage and Margin amount as numbers.
- 5 The Detail section should also include the ability to make individual selections based on values in the following fields:
 - Year
 - Month
 - CustomerCountry
 - Customer
 - Product
 - OrderID
 - OrderDate
 - Salesperson

The Sales Analysis should also show what selections have been made (the current selections) at all times.

Extra Credit

For extra credit, additional objects can be created, such as the compelling use of:

- Buttons
- Bookmarks
- Reports
- Text Objects
- Other objects, as appropriate

As well as the creative application of QlikView design best practices.

Key Measurements

The Company has defined the following Key Performance Indicators (KPIs) for the analysis application. Visibility to these measurements will indicate compliance with business goals through the various ways they are presented in the analysis tool (Dashboard, Sales and Details). These KPIs should have the ability to be trended over time.

Sales

Sales is defined as the total of the Sales field and can be expressed as

Sum(Sales)

Margin

Margin is defined as the total of the Margin field and can be expressed as

Sum(Margin)

Margin Percent

Margin Percent is defined as the percentage of Sales comprised of Margin and can be expressed as

Sum(Margin) / Sum(Sales)

Orders

Orders is defined as the number of individual orders and can be expressed as

Count(DISTINCT OrderID)

Average Order Value

Average Order Value is defined as Sales divided by Orders and can be expressed as

Sum(Sales) / Count(DISTINCT OrderID)

Quantity

Quantity is defined as the total of the Quantity field and can be expressed as

Sum(Quantity)

Note: even though these expressions are provided for you, be prepared to explain them to your teacher.

Data Sources

Sales.xls

This is the primary data source for the application. The tables (sheets) are described below.

Orders

This table (sheet within the Sales.xls file) contains all the order information for the application. Please note that transformation of the OrderDate field to obtain useful Year, Month and Week values involves more than simply loading this table into QlikView using the table wizard.

Hint: be prepared to add transformations to your load script, as in:

```
Year(OrderDate) as Year,  
Month(OrderDate) as Month,  
Week(OrderDate) as Week,
```

Note, too, that the key field linking the Orders table to Customers is the CustomerID field. The key field linking the Orders table to the Employees table is the EmployeeID field.

Customers

This table (sheet within the Sales.xls file) contains the all the customer information for the application. It is linked to the Orders table via the CustomerID field.

Employee.xls

This spreadsheet contains employee information for the application. Please note that additional desktop data script editing will be required to rename the fields as they are loaded from the Excel file.

Hint: be prepared to rename the fields as follows:

**EmpID as EmployeeID,
EmployeeName as Salesrep**

The Employees table links to the Orders table via the EmployeeID field. Since QlikView automatically associates (links) when field names are identical, we renamed the EmpID field, above, to match the value (EmployeeID) in the Orders table. We also renamed the EmployeeName field to Salesrep to make it more meaningful in our application.

Application Requirements

Here are the basic requirements that must be included within the application:

- Check the numbers to be sure they tie out with the source data
- Use a clean, consistent layout
- Pay attention to the proper use of color
- Implement your design best practices knowledge (from QlikView Designer I training)
- Build the three main analysis sheets: Dashboard, Sales and Details per the requirements, above
- Add at least two of the Extra Credit items and/or create additional charts and tables to support the business requirements and enhance the analytical power of the application
- Leverage resources, QlikCommunity, Qlikview.com, Best Practices documents, e-Learning, QlikOpedia

- Your data structure should look like this in the Table Viewer of your final application —



Figure 41. Table Viewer from the exercise once data has been loaded properly from the Excel source files

Note: There are undoubtedly many other options to improve the application. Your instructor will work with you to be sure you have mastered the assignment.

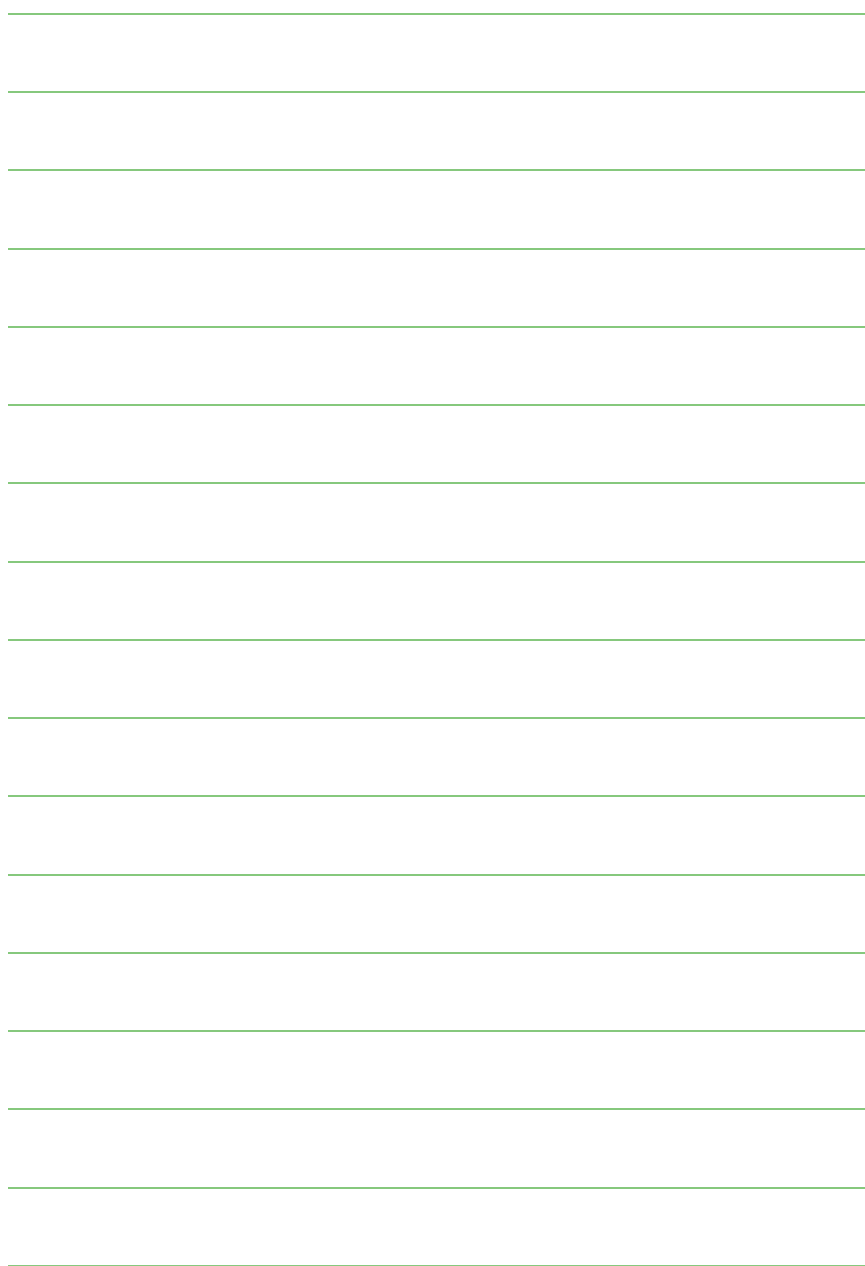
Final Comments on Building With QlikView

Here is some friendly QlikView designer advice:

- Copy
- Borrow
- Duplicate objects and change the dimensions/expressions

Remember, it is almost always better to re-use something, even if it originated in a completely different QlikView file or for a totally different purpose, than to build it from scratch.

Tip: it is almost always better to re-use something than to build it from scratch.



12 REVIEW AND CONCLUSION

Review

Time permitting, the class will review the business case applications both in terms of their success at meeting the business requirements and in using the design best practices and criteria identified earlier in the course. This will be done as a group activity with the assistance of the instructor.

Collaboration

Collaboration is one of the most important capabilities of real-world QlikView deployments. This topic is beyond the scope of this course, but, time permitting, your instructor might provide a demonstration.

Conclusion

QlikView Desktop is the power user's toolkit for creating compelling QlikView layouts and design. We sincerely hope this course has expanded your capabilities and confidence with QlikView.

Please take a moment to fill out a course evaluation form, or, if you would prefer, visit us online and complete an evaluation or send recommendations through the links available there.

